

PATENT ABSTRACTS OF JAPAN

#2

(11)Publication number : 10-093961

(43)Date of publication of application : 10.04.1998

(51)Int.Cl.

H04N 7/24

(21)Application number : 09-235398

(71)Applicant : SAMSUNG ELECTRON CO LTD

(22)Date of filing : 15.08.1997

(72)Inventor : READER CLIFF
SON JAE CHEOL
QURESHI AMJAD
NGUYEN LE

(30)Priority

Priority number : 96 699303
96 699382Priority date : 19.08.1996
19.08.1996

Priority country : US

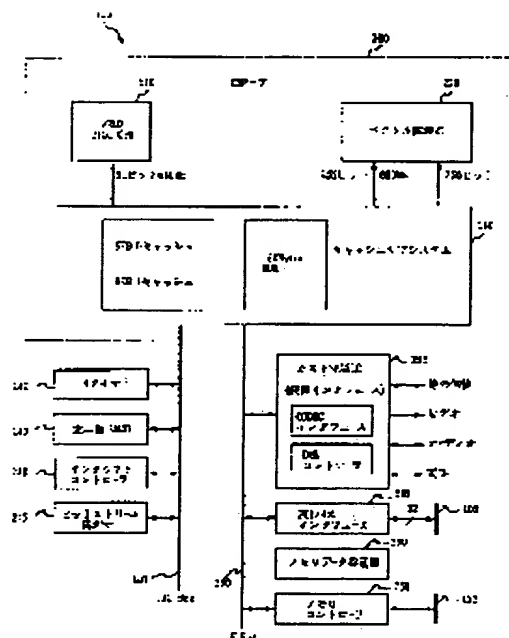
US

(54) METHOD FOR PROCESSING VIDEO DATA AND ITS DEVICE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a method and a circuit which can operate fast processing of video data.

SOLUTION: This device contains three processors, i.e., a scalar processor 210, a vector processor 220 and a bit stream processor 245. In encoding and decoding of video data, the processor 220 executes an operation that can efficiently be carried out by a single instruction multiple data processor. The processor 245 executes Huffman and RLC encoding or decoding. In the processor 245, a computer system processes simultaneously many data streams. The processors 210 and 220 can be programmed to execute a single arithmetics or a Boolean instruction. The processor 245 can be programmed to execute entire video data processing operations.



LEGAL STATUS

[Date of request for examination]

25.12.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the
examiner's decision of rejection or application
converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-93961

(43) 公開日 平成10年(1998) 4月10日

(51) Int.Cl.⁶

H 0 4 N 7/24

識別記号

F I

H 0 4 N 7/13

Z

審査請求 未請求 請求項の数 6 F D (全 129 頁)

(21) 出願番号 特願平9-235398

(22) 出願日 平成9年(1997) 8月15日

(31) 優先権主張番号 6 9 9, 3 0 3

(32) 優先日 1996年 8月19日

(33) 優先権主張国 米国 (U S)

(31) 優先権主張番号 6 9 9, 3 8 2

(32) 優先日 1996年 8月19日

(33) 優先権主張国 米国 (U S)

(71) 出願人 390019839

三星電子株式会社

大韓民国京畿道水原市八達区梅灘洞416

(72) 発明者 クリフ・リーダー

アメリカ合衆国・カリフォルニア・

95070・サラトガ・ビーオーボックス・
2666

(72) 発明者 ジャエ・チョル・ソン

アメリカ合衆国・カリフォルニア・

95014・クパティノ・パークウッド・ドラ
イヴ・# 1・10250

(74) 代理人 弁理士 志賀 正武 (外 2 名)

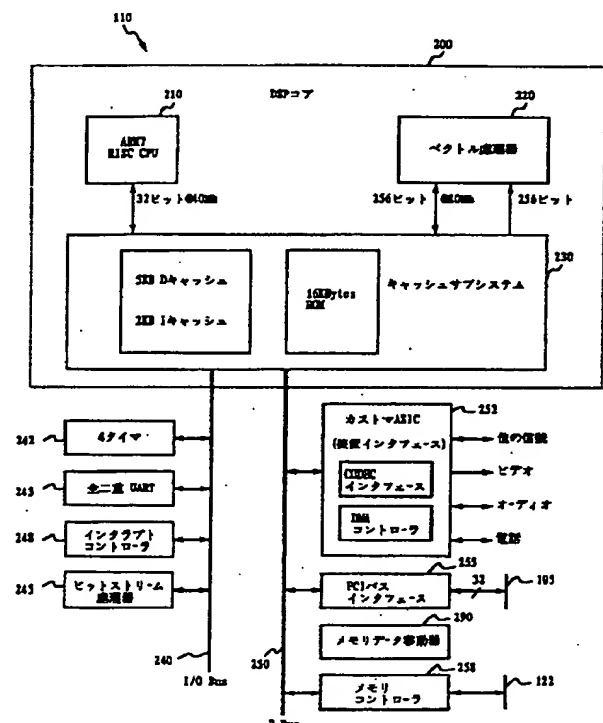
最終頁に続く

(54) 【発明の名称】 ビデオデータ処理方法及び装置

(57) 【要約】

【課題】 ビデオデータの高速処理動作ができる方法及び回路を提供する。

【解決手段】 同時に動作可能な3個の処理器、スカラー処理器210、ベクトル処理器220及びビットストリーム処理器245とを含む。ビデオデータをエンコーディングまたはデコーディングすることにおいて、ベクトル処理器220は、単一命令多重データ処理器により効率的に遂行され得る動作を遂行する。ビットストリーム処理器245は、ハフマン及びRLCエンコーディングまたはデコーディングを遂行する。ビットストリーム処理器245は、コンピュータシステムが多数個のデータストリームを同時に処理する。スカラー処理器210及びベクトル処理器220は、単一算術またはブール命令を実行するようにプログラムされ得る。ビットストリーム処理器245は、全体的なビデオデータ処理動作を遂行するためにプログラムされ得る。



【特許請求の範囲】

【請求項1】 ビデオデータをエンコーディングまたはデコーディングするためのシステムにおいて、ビデオデータに対して線形変換を遂行するベクトル処理器、

前記ベクトル処理器の出力を圧縮したり、前記のベクトル処理器に入力するためにビデオデータを復元するビットストリーム処理器、及び、

前記のベクトル処理器と前記ビットストリーム処理器の動作を同期化させる制御回路を含んでおり；前記のビットストリーム処理器は、1つのビデオデータストリームに対する処理を中止し、他のビデオデータストリームに対する処理を開始するように、前記の制御回路によりインタラプトされ、前記のシステムが2個のビデオデータストリームを実時間的にエンコーディングまたはデコーディングできるように、前記のビットストリーム処理器が2個のビデオデータストリームをほぼ同時に処理することができることを特徴とするビデオデータのエンコーディングまたはデコーディングシステム。

【請求項2】 前記の各ビデオデータストリームは、動画像を表わすことを特徴とする請求項1記載のビデオデータのエンコーディングまたはデコーディングシステム。

【請求項3】 ビデオデータをエンコーディングまたはデコーディングするためのシステムにおいて、ビデオデータに対して線形変換を遂行するベクトル処理器、及び前記ベクトル処理器の出力を圧縮したり、前記のベクトル処理器に入力するためにビデオデータを復元するビットストリーム処理器を含んでおり；前記のベクトル処理器は、単一算術またはブール命令を実行できるようにプログラムされ得、前記のビットストリーム処理器は、単一算術またはブール命令を実行できるようにプログラムされ得ないことを特徴とするビデオデータのエンコーディングまたはデコーディングシステム。

【請求項4】 ビデオデータをエンコーディングまたはデコーディングするための方法において、ビデオデータに対し線形変換を遂行するベクトル処理器、及び前記ベクトル処理器の出力を圧縮したり、前記のベクトル処理器に入力するためにビデオデータを復元するビットストリーム処理器を含んでおり；前記のベクトル処理器は、単一算術またはブール命令を実行できるようにプログラムされ得、前記のビットストリーム処理器は、単一算術またはブール命令を実行できるようにプログラムされ得ないことを特徴とするビデオデータのエンコーディングまたはデコーディング方法。

【請求項5】 前記の各ビデオデータストリームは、動画像を表わすことを特徴とする請求項4記載のビデオデータのエンコーディングまたはデコーディング方法。

【請求項6】 ビデオデータをエンコーディングまたはデコーディングするための方法において、

ビデオデータに対し線形変換を遂行するベクトル処理器、及び前記ベクトル処理器の出力を圧縮したり、前記のベクトル処理器に入力するためにビデオデータを復元するビットストリーム処理器を含んでおり；前記のベクトル処理器は、単一算術またはブール命令を実行できるようにプログラムされ得、前記のビットストリーム処理器は、単一算術またはブール命令を実行できるようにプログラムされ得ないことを特徴とするビデオデータのエンコーディングまたはデコーディング方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、コンピュータによるデータ処理に関し、特にコンピュータによるビデオデータ処理に関する。

【0002】

【従来の技術】 通常、コンピュータはシステムデータを圧縮したり復元するために使用されて来た。システムデータには、停止及び／または動画像のイメージを含むビデオデータが含まれる。また、システムデータには、オーディオデータ、例えば動画像のサウンドトラックが含まれる。ビデオデータの高速処理が可能な方法及び回路を提供することが好ましい。

【0003】

【発明が解決しようとする課題】 従って、本発明の目的は、ビデオデータの高速処理ができる方法及び回路を提供することにある。

【0004】

【課題を解決するための手段】 いくつかの実施例において、本発明によるコンピュータシステムは、同時に動作可能な3個の処理器、すなわちスカラー処理器、ベクトル処理器及びビットストリーム処理器を含む。ビデオデータをエンコーディングまたはデコーディングすることにおいて、ベクトル処理器は、単一命令多重データ(Single Instruction Multiple Data: SIMD) 処理器により効率的に行われる動作を遂行する。このような動作としては、1) 離散余弦変換(Discrete Cosine Transform: DCT) のような線形データ変換、2) モーション補償などがある。ビットストリーム処理器は、ワードまたは半ワード(half-words)より特定のビット上における動作を含む動作を遂行する。このような動作としては、例えば、MPEG-1、MPEG-2、H. 261、H. 263に使われるハフマン(huffman)及びRLCエンコーディングとデコーディングなどがある。スカラー処理器は、ハイレベルビデオ処理(例えば、ピクチャーレベル処理)を遂行し、ベクトル及びビットストリーム処理器の動作を同期化させ、外部装置のインターフェースを制御する。いくつかの実施例において、コンピュータシステムは、多数のデータストリームを同時に処理することができる。その結果、コンピュータシステムの利用者は、2個以上の会合や画像会議も可能である。ビットス

トリーム処理器では、多様なビットストリームが実時間的に同時にエンコーディングまたはデコーディングされるように文脈(contexts)を切換えることができるので、多重データストリームの同時処理が可能になる。

【0005】いくつかの実施例において、スカラー及びベクトル処理器は、各処理器が単一算術命令またはブール(boolean)命令を遂行するようにプログラムされ得る点から見るとプログラム可能である。ビットストリーム処理器は単一算術命令またはブール(boolean)命令を遂行するようにプログラムされ得ない点から見ると、プログラム不能である。むしろ、ビットストリーム処理器は、1セットのビデオデータに対し、全体的なビデオデータ処理動作を遂行するようにプログラムされ得る。ビットストリーム処理器が、単一算術命令またはブール命令を遂行するためにプログラムされないようにすることで、ビットストリーム処理器が高速で動作することができる。スカラー及びベクトル処理器がプログラム可能にすることにより、ビデオデータエンコーディング及びデコーディングの標準から変更されたシステムを採択することが容易である。

【0006】

【発明の実施の形態】図1は、マルチメディア処理器110を含むメディアカード100を示している。この実施の形態において、マルチメディア処理器110は、その仕様がカリフォルニアサンホセに常住する三星半導体株式会社で製作されるタイプMSP-1EX(商標名)処理器である。処理器MSP-1EXは下記の付録Aに記述されている。処理器110は、ローカルバス105を通してホストコンピュータシステム(図示せず)と通信する。いくつかの実施例において、バス105は、32ビット、33MHz PCIバスである。処理器110から出力されるデジタルビデオデータは、D/A(デジタル/アナログ)変換器112に結合される。ビデオ部分だけでなく、デジタルビデオデータは、オーディオ部分、例えば映画のサウンドトラックを含むことができる。変換器112の出力は、アナログデータを処理するTVセット(図示せず)または他のシステムに結合され得る。いくつかの実施例において、処理器110はA/D(アナログ/デジタル)変換器(図4から図6参照)から出力されるデジタルビデオデータを受信するための入力ポートを含む。

【0007】処理器110は、コーデック(CODEC)114に連結される。CODEC114は、テープレコーダ(図示されていない)または他の装置からアナログオーディオデータを受信する。CODEC114は、電話線(図示されていない)からアナログ電話データを受信する。CODEC114は、アナログデータをデジタル化してから、これを処理器110へ伝送する。CODEC114は、処理器110からデジタルデータを受信し、このデータをアナログ形態に変換し、必要に

よってこのアナログデータを伝送する。処理器110はバス122によりメモリ120に連結される。図1において、メモリ120はSDRAM(synchronous DRAM)であり、バス122は64ビット、80MHzバスである。他の実施例では、他のメモリ、バス幅、及びバス速度が使用される。非同期メモリ及びバスがいくつかの実施例に使用される。カード100のいくつかの実施例は、Le. Nguyenを出願人とし、本出願と同日付で出願された“Multiprocessor Operation in a Multimedia Signal Processor”という発明の名称を有する、米合衆国特許出願明細書(弁理士参照番号:M-4364US)に記載されており、その全体的な内容は、本発明で参照として引用される。

【0008】図2は、処理器110の一実施例によるブロック図である。処理器110は、スカラー処理器210、ベクトル処理器(VP)220及びビットストリーム処理器(BP)245を含む。いくつかの実施例において、処理器210は40MHzで動作し、公知の標準ARM7命令語セットを支援する32ビットRISC処理器である。ベクトル処理器220は80MHzで動作し、288ビットベクトルレジスタを備えた単一命令多重データ(SIMD)処理器である。VP220の一実施例は、Songらを出願人とし、本出願と同日付で出願された“Efficient Context Saving and Restoring in a Multitasking Computing System Environment”という発明の名称を有する、米合衆国特許出願明細書(弁理士参照番号:M-4365US)に記載されており、その全体的な内容は、本発明で参照として引用される。処理器210、220は、単一算術命令またはブール命令または、これらの命令のシーケンスを遂行するようにプログラムできる。

【0009】いくつかの実施例において、ビデオデータ処理を高速で行うためにビットストリーム処理器245は、単一算術命令またはブール命令を遂行するためにプログラムされないように設計される。特に、BP245はADD、OR、“ADDAND ACCUMULATE”等のような単一命令を遂行するようにプログラムされ得ない。かえって、BP245は付録Aの10章に記述されているビデオデータ処理動作を遂行するようにプログラムされる。これと同時に、スカラー処理器210とベクトル処理器220は、単一算術またはブール命令を遂行するようにプログラムされ得る。従って、処理器110は、ビデオ標準から変形を図ることができる。

【0010】図2の図示のとおり、スカラー処理器210とベクトル処理器220は、キャッシュサブシステム230に連結される。キャッシュサブシステム230は、バス(IOBUS;240)とバス(FBUS;250)に連結される。いくつかの実施例において、IOBUS240は32ビット、40MHzバスであり、FBUS250は64ビット、80MHzである。IOBUS240は、ビ

ットストリーム処理器245、インタラプトコントローラ248、全二重通信(full-duplex)UARTユニット243と、4個のタイマー242に連結される。FBUS250は、メモリバス122(図1参照)に連結されたメモリコントローラ258に連結される。FBUS250は、PCIバス105に連結されたPCIバスインターフェース回路255に連結される。また、FBUS250は、ビデオD/A112(図1参照)、CODEC114と場合によってビデオA/D変換器(図4から図6の図示と同様である)をインターフェイスする回路を含む、装置インターフェース回路252("Customer ASIC"とも呼ばれる)に連結される。また、処理器110はメモリデータ移動器290を含む。

【0011】処理器110は、多数個のデータストリームが同時に処理可能である。例えば、処理器110の利用者が2個以上の会合と画像会議をする場合、処理器110は利用者が多数個の会合が視聴できるように、ビデオ及びオーディオ処理を遂行する。多重ビデオデータストリームを処理するために、処理器110は文脈切換を支援する。これはBP245が多重データストリームの間を切換することを意味する。画像会議において、各データストリームは、遠く離れている別個の会合から送られることもできる。その代案として、利用者が映像会議に参加し、同時に画像会議または映画上映を視聴することができるように、付加的なデータストリームが映画チャンネルから送られることができる。文脈切換は、実施例の10.12節に記述されている。文脈が切り換えられると、スカラー処理器210は現在の文脈を貯蔵し、他の文脈を処理するためにBP245を初期化させる。

【0012】BP245は、下記のようなビデオデータフォーマットすなわち、

1. ISO/IEC標準11172(1992年)に記述されているMPEG-1;
 2. 文書ISO/IEC JTC 1/SC 29 N 0 981 Rev(1995年3月31日)に記述されるMPEG-2;
 3. "ITU-T勧告H. 261"(1993年3月)に記述されているH. 261;及び
 4. "ドラフトITU-T勧告H. 263"(1996年5月2日)に記述されているH. 263
- を処理することができる。

【0013】ビデオデータは、スカラー処理器210、ベクトル処理器220及びビットストリーム処理器245に分けられて処理されることによって、高速処理が実現される。さらに詳しくは、ベクトル処理器220は、線形変換(DCTまたは逆DCT)とモーション補償を遂行する。このような動作は、ベクトル処理器に適する。なぜなら、これらの動作は、時々データのいろんな部分に対して遂行される同一の命令を必要とするからである。ビットストリーム処理器245は、ハフマン

デコーディング及びエンコーディングとジグザグビットストリーム処理を遂行する。スカラー処理器210は、ビデオ及びオーディオ逆多重化と同期化及びI/Oインターフェーシング作業を遂行する。エンコーディング及びデコーディング動作の例は、実施例1の10.6.1及び10.6.2節に現れている。エンコーディング動作において、圧縮されないデジタルデータが、バス105を通してフレームメモリ120またはホストシステム(図示せず)から到着する。いくつかの実施例において、装置インターフェース回路252は、ビデオA/D変換器を含み、圧縮されていないデータが変換器から到着する。ベクトル処理器220は、量子化、DCT及びモーション補償を遂行する。ビットストリーム処理器245は、VP220の出力を受信し、GOB(Group of Blocks)及びスライスを生成する。特に、BP245はハフマン及びRLCエンコーディングとジグザグビットストリーム処理を遂行する。スカラー処理器210は、BP245の出力を受信し、ピクチャー階層符号化(picture layer coding)、GOP(group of pictures)符号化及び、シーケンス階層符号化を遂行する。その後、スカラー処理器210は、オーディオ及びビデオデータを多重化し、符号化されたデータをバス(105または122)を通して、貯蔵装置またはネットワークに伝送する。ネットワークへの伝送は、いくつかの実施例のネットワークに連結された装置インターフェース回路252への伝送を含む。

【0014】デコーディングにおいて、処理は逆に遂行される。スカラー処理器210は、システムデータをビデオ及びオーディオ成分に逆多重化し、ビデオデータのシーケンス階層、GOP及びピクチャー階層デコーディングを遂行する。その結果、生成されたGOBまたはスライスは、ビットストリーム処理器245に供給される。処理器245はジグザグ処理とハフマン及びRLCデコーディングを遂行する。VP220は、BP245の出力を受信し逆量子化、IDCT及びモーション補償を遂行する。VP220は、必要なら(例えば、ピクチャーイメージのエッジを平坦化させる場合)、任意の前処理を遂行し、復元されたデジタルピクチャーを装置インターフェース回路252または貯蔵装置に供給する。スカラー処理器210、ベクトル処理器220とビットストリーム処理器245は、多くのブロックのデータに対して並列に動作することができる。スカラー処理器210がピクチャー層及び上位層を処理することにより、処理器内部の通信を減少させる。これは、ピクチャー層及び上位層が、制御及びI/O機能のためにスカラー処理器210では使用されるが、ベクトル処理器220及びビットストリーム処理器245では使用されない情報を含んでいるからである。このような情報の例としては、フレームを装置インターフェース回路252に伝送するために、スカラー処理器210で使用されるフレー

ムレートが挙げられる。

【0015】図3は、ビットストリーム処理器245の一実施例によるブロック図である。図3に示された信号は、実施例1の10.5節に記述されている。この信号は、ビットストリーム処理器245とIOBUS240（図2参照）との間のインターフェースを提供する。BP245において、これらの信号は、SRAM320を含むIOBUSインターフェースユニット310により処理される。また、BP245は、VLC FIFOユニット330、VLC LUT ROM340、制御ステートマシン350と、レジスタファイルとSRAMを含むBPコアユニット360を含む。図3のブロックは、実施例1の10.4節に記述されている。ROM340は、4個の標準、すなわちMPEG-1、MPEG-2、H.261及びH.263に対しハフマンエンコーディング及びデコーディング時に使用されるルックアップテーブルを含む。テーブルに貯蔵される情報量が膨大であるにも拘わらず、ROM340は768*12ビットの小さいサイズを有する。小さいサイズはテーブルを共有し、実施例1の4節に記述されているような他の技術により実現される。

【0016】本発明を特定の好ましい実施例に関連し図示し説明したが、本発明はそれに限定されず、特許請求の範囲により備えられる精神や分野から離脱しない限度内で、本発明が多様に改造及び変化され得ることが、当業界で通常の知識を有する者なら容易に分かる。特に、本発明は任意の回路、クロックレートまたはこれらの実施例のタイミングにより限定されるものでない。

【0017】

【実施例1】

MSP-1EXシステム仕様

第1章 技術的な概要

本章は、ハードウェア及びソフトウェア設計者が見せあげるマルチメディア信号処理器（“MSP-x”）の技術的な概要を説明する。

1.1 機能

マルチメディア信号処理器（MSP-x）は、パーソナルコンピュータ及び注文者製品応用のための広範囲な集積機能を提供するために、一群の単一チップVLSI装置を形成する。MSP群は、最適の費用/性能のために、計算に対する単一命令多重データ（SIMD）モデルを適用する強力なベクトル処理器の構造に基づいたものである。その特性は下記のとおりである。

* 完全なプログラム可能性

* ARM命令語セット構造に基づく。

* 集積された40MHz ARM7 RISC CPUコア

* 高性能デジタル信号処理のための80MHzベクトル処理器

* 9ビット整数ALU動作のための2.56Gops 50

* 16ビット整数掛け算-累積動作のための2.56Gops

* 32ビットIEEE浮動小数点加算のための640Mflops

* 32ビットIEEE浮動小数点掛け算&加算のための1280Mflop

* 選択的な注文型化またはグラフィックス機能のための未使用の10K gates

* 0.65μm 3.3v/5v CMOS技術に基づく。

* 128ピン-128ピンパッケージ
MSPは初期に4個の主要機能を支援する。

* ビデオ

* オーディオ/サウンド

* 遠距離通信

* 2D/3Dグラフィックス（選択）

1.1.1 ビデオ

* 全機能がファームウェアでプログラム可能である。

* 実時間MPEG-1デコーディング及びエンコーディング

* 実時間MPEG-2デコーディング

* ほぼ実時間的なMPEG-2エンコーディング

* 実時間H.324デコーディング及びエンコーディング

* 任意のスクリーンサイズまたは解像度に対するイメージスケール

* RGBとYUV間の色空間変換

* ピクチャー輪郭強調及び雑音減少のためのイメージフィルタリング

30 * 4/3フルダウ変換

1.1.2 オーディオ/サウンド

* 全機能がファームウェアでプログラム可能である。

* 実時間MPEG-1オーディオデコーディング及びエンコーディング

* 実時間MPEG-2オーディオデコーディング及びエンコーディング

* 実時間H.320及びH.324オーディオデコーディング及びエンコーディング

* 実時間G.728及びG.723音声コーディング

* 実時間サウンドブラスターエミュレーション

* ウェーブテーブル合成

* FM合成

1.1.3 遠距離通信

1.1.3.1 モデム

* 標準非同期COMポートインターフェース（NS16550A UART互換可能）

* 28.8Kから2.4KbpsまでのV.34

* 4800、9600無符号化及び9600bpsトレリス符号化に対するデータレート有するCCITT-V.32bis

* Hayes AT命令語セットの互換性

* 呼出進行モニタ

* V. 25bisオートダイヤル

* DTMF及びパルスダイアリング

* 非同期エラー復旧プロトコル

* V. 42エラー訂正

1. 1. 3. 2 ファクシミリ

* 9600bpsまたは7200bpsのV. 29

* 4800bpsまたは2400bpsのV. 27

* 呼出進行モニタ

* DTMF及びパルスダイアリング

* G3トランスファーク

* T. 4/T. 30動作

1. 1. 3. 3 電話応答

* 電話機セットまたはマイクロフォンを通して挨拶の言葉録音

* 受信された電話に対し自動応答し、予め録音されたメッセージに応信

* 電話をかけた相手からのメッセージ録音

* 電話をかけた相手が残したメッセージ再生

1. 1. 4. 2D/3Dグラフィックス (選択)

* BITBLT

* 2Dライン&多角形ドローイング及びシェーディング

* 3Dポイント、ライン及び三角形に対する幾何学及び採光計算

* テクスチャマッピングで3Dカラー計算

* プレンディング

【0018】1. 2 ハードウェアの構造

1. 2. 1 概要

MSP-1マルチメディアコプロセッサ群は、集積度レベル、費用及び性能を含む多様な要求事項を満足させるように設計する。MSP-1処理器を含むブロック図は図4の図示のとおりである。MSP-1群は、下記のようなピンアウトオプションを行う。

* MSP-1は、外部SDRAMを使用せず、エントリレベルとして使用されるように設計される。

* MSP-1EXは、外部SDRAMとインターフェーシングのための32ビットメモリを含む。

* MSP-1Fは、外部SDRAMとインターフェーシングのための64ビットメモリを含む。

* MSP-1Gは、集積されたSVGAコントローラ、高速化した3Dグラフィックス加速が加えられたRAMDACを含む。図5は、MSP-1E処理器を含むシステムのブロック図である。

1. 2. 2 外部コーデック

図6は、外部コーデックと共にMSP-1処理器を含むシステムのブロック図である。

【0019】1. 2. 2. 1 MSP-1EXの材料目録

次は、MSP-1EXに対して提示された材料目録であ

る。

* MSP-1EX

* 512K×32ビット同期DRAM

* NTSC/PALエンコーダー (三星のKS0119)

* オーディオ&遠距離通信CODEC (アナログデバイス社のAD1843)

* その他 (キャパシタ、抵抗、増幅器、コネクタ等)

* プリントされた回路基板

10 【0020】1. 3 マイクロ構造

1. 3. 1 概要

基本的にMSPマイクロ構造は、非常に強力なDSPコアと、注文者社により規定されたメモリ&I/Oサブシステムとから構成される (図2参照)。DSPコアは、下記のことを含む。

* 40MHzで動作し、一般的な処理のために使用される32ビットARM7 RISC CPU

* 80MHzで動作し、信号処理のために使用されるベクトル処理器

20 * 80MHzで動作し、2KB命令キャッシュ、5KBデータキャッシュ及び16KB ROMキャッシュを有する共有されたキャッシュサブシステム。データキャッシュは、ハードウェアまたはソフトウェアにより制御され得る。

* 80MHzで動作し、内部の多くのFBUS周辺機器とインターフェースする高速64ビットバス (FBUS)

* 40MHzで動作し、内部の多くのIOBUS周辺機器とインターフェースする低速32ビットバス (IOBUS)

内部のFBUS周辺機器は下記のものを含む。

* 32ビット33MHz PCIバスインターフェース

* 64ビットSDRAMメモリコントローラ

* 8チャンネルDMAコントローラ

* 注文者ASICロジックブロック。注文者ASICロジックブロックは、多様なアナログコーデックに対するインターフェースと、注文者の規定したI/O装置を含む合計10K gatesを提供する。インターフェースロジックは、三星のKS0119NTSCエンコーダー及び、アナログデバイス社のAD1843コーデックを支援する。

【0021】* ホスト (Pentium) メモリからMSPローカルSDRAMメモリまでのデータをDMAすることに使用されるメモリデータ移動器

* ビデオビットストリームを処理するビットストリーム処理器

* 16450 UART シリアルライン

* 8254 - 互換可能なタイマー

* 8259 - 互換可能なインタラプトコントローラ

また、MSPはソフトウェアで制御される初期化及び、

インタラプトのために使用される特殊なレジスタ (MS * 【0022】
P制御レジスタ) を含む。

*

1. 4 MSP-1EXピン説明

1. 4. 1 合計: 256ピン

1. 4. 2 PCIバスインターフェース (53ピン)

CLK	クロック入力ピン
RSTL	入力ピンリセット、アクチブロー
AD[31:0]	アドレス及びデータバスピン
C_BE0L	コントロール&バイト0イネーブルピン、アクチブロー
C_BE1L	コントロール&バイト1イネーブルピン、アクチブロー
C_BE2L	コントロール&バイト2イネーブルピン、アクチブロー
C_BE3L	コントロール&バイト3イネーブルピン、アクチブロー
PAR	パリティピン
FRAMEL	サイクルフレームピン、アクチブロー
IRDYL	開始者準備ピン、アクチブロー
TRDYL	ターゲット準備ピン、アクチブロー
STOPL	停止トランザクションピン、アクチブロー
LOCKL	ロックトランザクションピン、アクチブロー
IDSEL	初期化装置選択入力ピン
DEVSEL	装置選択ピン、アクチブロー
REQL	バス要請ピン、アクチブロー
GNTL	バス承認ピン、アクチブロー
PERRL	パリティエラーピン、アクチブロー
SERRL	システムエラーピン、アクチブロー
INTAL	インタラプトピン、アクチブロー

1. 4. 3 その他 (6ピン)

TCK	JTAGテストクロック入力ピン
TDI	JTAGテストデータ入力ピン
TDO	JTAGテストデータ出力ピン
TMS	JTAGテストモード選択入力ピン
TRSTL	JTAGテストリセット入力ピン
CLK	クロック入力。これは40MHzクロック入力ピンである。

1. 4. 4 KS0119 NTSC/PALエンコーダーインターフェース (24ピン)

SFRS	3ワイヤーホストインターフェースのためにKS0119に出力されるフレーム同期
SCLK	KS0119に出力されるシリアルクロック
SDAT	シリアルデータI/O
BGHS	MSPに入力される水平同期信号
BGVS	MSPに入力される垂直同期信号
MSSEL	マスタ選択
PD[15:0]	KS0119に出力されるピクセルデータ
BGCLK	KS0119に出力されるピクセルクロック
PROMCSL	BIOS PROMチップ選択

【0023】1. 4. 5 AD1843オーディオ&遠 距離通信コーデックインターフェース (6ピン)

A43SCLK シリアルクロック入/出力。SCLKはバスマスタ (BM) ピンがHIに駆動される場合、クロックをシリアルバスに対する出力として供給し、B

MインがLOに駆動される場合、クロックを入力として受け入れる両方向信号である。

A43SDFS シリアルデータフレーム同期入/出力。SDFSはバスマスタ (BM) ピンがHIに駆動される場合、フレーム同期信号をシリアルバスに対する出

力として供給し、BMピンがLOに駆動される場合、フレーム同期信号を入力として受け入れる両方向信号である。

A43SDI MSPから出力されるAD1843に対するシリアルデータ入力。全制御及び再生トランスファーは、16ビット長さのMSBである。

A43SDO AD1843から出力されMSPに入力されるシリアルデータ出力。全ステータス&制御レジスタ読出及び再生トランスファーは、16ビット長さのMSBである。

【0024】1. 4. 6 メモリバスインターフェース
(87ピン)

RAS1L 出力ピン (アクチブロー)。これはMA[11:0]からのローアドレスを、選択されたSDRAMバンクの内部ローアドレスバッファにラッチするローアドレスストローブである。

CAS1L 出力ピン (アクチブロー)。これはMA[11:0]からのコラムアドレスを、選択されたSDRAMバンクの内部コラムアドレスバッファにラッチするコラムアドレスストローブである。

MWEL 出力ピン (アクチブロー)。これはSDRAMに対する記入イネーブルである。

MAI[11:0] 出力ピン。SDRAMに対し多重化されたロー及びコラムアドレス信号。

MD[63:0] 入/出力SDRAMデータピン *

* MA23 出力ピン。メモリアドレスビット<23>

MA24 出力ピン。メモリアドレスビット<24>

DQM 出力ピン。クロック以降、SDRAMデータをハイインピーダンスにし、出力をマスクさせる。(このピンは、同期DRAMインターフェースためにのみ使用される。)

10 MCKE 出力ピン。次のクロックサイクルから動作を中止させるために、SDRAMシステムクロックをマスクさせる。

MCS0L 出力ピン (アクチブロー)。下位32ビットに対するSDRAMチップ選択

MCS1L 出力ピン (アクチブロー)。上位32ビットに対するSDRAMチップ選択

MR. DYH 出力ピン。SDRAM準備信号

MEMCLK 出力ピン。これはSDRAMに対するクロック出力ピンである。

1. 4. 7 電源

20 VDD 3.3ボルト電源ピン

VCC 5ボルト電源ピン

VSS 接地ピン

MSP-1EXピン指定

【0025】

【表1】

MSP-1EXピン指定

チップパッド#	パッケージピン#	タイプ	名称	説明
1	202	IN	VDD	3.3V電源供給
2	1	I/O	AD31	PCIバス用 アドレス及び データ
3	2	I/O	AD30	
4	3	I/O	AD29	
5	4	I/O	AD28	
6	5	I/O	AD27	
7	205	IN	GND	接地
8	6	IN	VCC	5V電源供給
9	7	I/O	AD26	アドレス及びデータ
10	8	I/O	AD25	PCIバス用 アドレス及び データ出力
11	9	I/O	AD24 -S08	アドレス及びデータ 出力テストシリアル
12	10	I/O	C_BE3L	PCIバス用 アドレス及び データ出力
13	208	IN	VDD	3.3V電源供給
14	11	IN	IDSEL	PCI用初期化選択
15	12	I/O	AD23 S07	PCIバス用
16	13	I/O	AD22 S06	アドレス及びデータ
17	14	I/O	AD21 S05	アドレス及びデータ /テストシリアル 出力
18	15	I/O	AD20 S04	出力
19	209	IN	GND	接地
20	16	IN	VCC	電源供給
21	17	I/O	AD19 S03	PCIバス用アドレ ス及びデータ/
22	18	I/O	AD18 S02	テストシリアル出力
23	19	I/O	AD17 S01	出力
24	210	IN	VDD	3.3V電源供給
25	20	I/O	AD16 S00	PCIバス用アドレ ス及びデータ出力
26	21	I/O	C_BE2L	PCIバス用 アドレス及び データ出力
27	22	I/O	FRAMEL	PCIバス用 アドレス及び データ出力
28	23	I/O	IRDYL	PCIバス用 アドレス及び データ出力
29	211	IN	GND	接地
30	24	I/O	TRDYL	PCIバス用 アドレス及び データ出力
31	25	IN	VCC	5V電源供給
32	26	I/O	DVSELL	PCIのための 電源選択
33	27	I/O	STOPL	PCIに対する要求 を中止する
34	28	I/O	LOCKL	PCIに対する要求 をロックする

【表2】

チップパッド#	パッケージピン#	タイプ	名称	説明
35	214	IN	VDD	3.3V電源供給
36	29	I/O	PERRL	PCIパリティ
37	30	I/O	SERRL	PCIシステム
38	31	IN	TCA	テストクロックA
39	32	I/O	PAR	PCIのパリティ
40	215	IN	GND	接地
41	33	I/O	C_BE1L	PCI用 バイトネーグル
42	34	I/O	AD15_S19	PCIバス用アドレ ス及びデータ テストシリアル入力
43	35	IN	VCC	5V電源供給
44	36	I/O	AD14_S18	PCIバス用アドレ ス及びデータ テストシリアル入力
45	37	I/O	AD13_S17	
46	38	I/O	AD12_S16	
47	218	IN	VDD	3.3V電源供給
48	39	I/O	AD11_S15	PCIバス用
49	40	I/O	AD10_S14	アドレス及びデータ
50	41	I/O	AD09_S13	テストシリアル
51	42	I/O	AD08_S12	入力
52	221	IN	GND	接地
53	43	I/O	C_BE0L	PCI用 バイトネーグル
54	44	IN	TCB	テストクロックA
55	45	IN	VCC	電源供給
56	46	IN	TM	テストモード
57	47	I/O	AD07_S11	PCIバス用アドレ ス及びデータ テストシリアル入力
58	48	I/O	AD06_S10	PCIバス用アドレ ス及びデータ テストシリアル入力
59	222	IN	VDD	3.3V電源供給
60	49	I/O	AD05_MT5	PCIバス用アドレ ス及びデータ テストシリアル入力
61	50	I/O	AD04_MT4	PCIバス用アドレ ス及びデータ テストシリアル入力
62	226	IN	VDD	3.3V電源供給
63	227	IN	GND	接地
64	51	IN	VCC	5V電源供給

【表3】

チップパッド#	パッケージピン#	タイプ	名称	説明
65	52	I/O	AD03_MT3	PCバス用アドレスデータストレータ
66	53	I/O	AD02_MT2	PCバス用アドレスデータストレータ
67	227	IN	GND	接地
68	54	I/O	AD01_MT1	PCバス用アドレスデータストレータ
69	55	I/O	AD00_MT0	PCバス用アドレスデータストレータ
70	228	IN	VDD	3.3V電源供給
71	229	IN	GND	接地
72	56	OUT	MA11	SDRAM
73	57	OUT	MA10	
74	58	OUT	MA9	
75	59	OUT	MA8	
76	60	OUT	MA7	
77	230	IN	VDD	3.3V電源供給
78	231	IN	GND	接地
79	61	OUT	MA6	SDRAM
80	62	OUT	MA5	
81	63	OUT	MA4	
82	64	OUT	MA3	
83	232	IN	VDD	3.3V電源供給
84	233	IN	GND	接地
85	65	OUT	MA2	SDRAM
86	66	OUT	MA1	
87	67	OUT	MA0	
88	68	OUT	RAS1L	SDRAMストローク
89	234	IN	VDD	3.3V電源供給
90	235	IN	GND	接地
91	69	OUT	CAS1L	SDRAMアクセス
92	70	OUT	MEMCLK	SDRAMクロック
93	71	OUT	NWEL	SDRAMウェル
94	72	OUT	DQM	SDRAM DQM
95	73	OUT	MCSOL	SDRAMチップ(ローワード)
96	74	OUT	MCSOL	SDRAMチップ(ハイワード)

【表4】

チップ・ヘッド#	パッケージ・ピン#	タイプ	名 称	説 明
97	236	IN	VDD	3.3V電源供給
98	237	IN	GND	接地
99	76	OUT	MCKE	SDRAMクロック
100	76	IN	MRDYH	SDRAM準備
101	238	IN	VDD	3.3V電源供給
102	239	IN	GND	接地
103	77	I/O	MD0	SDRAM データバス
104	78	I/O	MD1	
105	79	I/O	MD2	
106	80	I/O	MD3	
107	240	IN	VDD	SDRAM
108	241	IN	GND	
109	81	I/O	MD4	
110	82	I/O	MD5	
111	83	I/O	MD6	SDRAM
112	84	I/O	MD7	
113	242	IN	VDD	
114	243	IN	GND	
115	85	I/O	MD8	SDRAM
116	86	I/O	MD9	
117	87	I/O	MD10	
118	88	I/O	MD11	
119	244	IN	VDD	3.3V電源供給
120	245	IN	GND	接地
121	89	I/O	MD12	SDRAM
122	90	I/O	MD13	
123	91	I/O	MD14	
124	92	I/O	MD15	
125	246	IN	VDD	3.3V電源供給
126	247	IN	GND	接地
127	93	I/O	MD16	SDRAM
128	94	I/O	MD17	
129	95	I/O	MD18	
130	96	I/O	MD19	
131	248	IN	VDD	3.3V電源供給
132	249	IN	GND	接地
133	97	I/O	MD20	SDRAM
134	98	I/O	MD21	
135	99	I/O	MD22	
136	100	I/O	MD23	

【表5】

チップ・ピン#	パッケージ・ピン#	タイプ	名称	説明
137	250	IN	VDD	3.3V電源供給
138	251	IN	GND	接地
139	101	I/O	MD24	SDRAM
140	102	I/O	MD25	
141	103	I/O	MD26	
142	104	I/O	MD27	
143	254	IN	VDD	3.3V電源供給
144	255	IN	GND	接地
145	105	I/O	MD28	SDRAM
146	106	I/O	MD29	
147	107	I/O	MD30	
148	108	I/O	MD31	
149	256	IN	VDD	3.3V電源供給
150	257	IN	GND	接地
151	109	I/O	MD32	
152	110	I/O	MD33	
153	111	I/O	MD34	
154	112	I/O	MD35	SDRAMデータバス
155	258	IN	VDD	3.3V電源供給
156	259	IN	GND	接地
157	113	I/O	MD36	SDRAM
158	114	I/O	MD37	
159	115	I/O	MD38	
160	116	I/O	MD39	
161	260	IN	VDD	3.3V電源供給
162	261	IN	GND	接地
163	117	I/O	MD40	
164	118	I/O	MD41	SDRAM
165	119	I/O	MD42	
166	120	IN	MSPCK	MSPシステム
167	262	IN	VDD	3.3V電源供給
168	263	IN	GND	接地
169	121	I/O	MD43	SDRAM
170	122	I/O	MD44	
171	123	I/O	MD45	
172	124	I/O	MD46	
173	264	IN	VDD	3.3V電源供給
174	265	IN	GND	接地
175	125	I/O	MD47	SDRAM
176	126	I/O	MD48	
177	127	I/O	MD49	
178	128	I/O	MD50	

【表6】

チップ番号	ピン番号	タイプ	名称	説明
179	266	IN	VDD	3.3V電源供給
180	267	IN	GND	接地
181	129	I/O	MD51	SDBAM
182	130	I/O	MD52	
183	131	I/O	MD53	
184	132	I/O	MD54	
185	269	IN	GND	接地
186	133	I/O	MD55	SDBAM
187	134	I/O	MD56	
188	135	I/O	MD57	
189	136	I/O	MD58	
190	270	IN	VDD	3.3V電源供給
191	271	IN	GND	接地
192	137	OUT	MD59	SDBAM
193	138	I/O	MD60	
194	139	I/O	MD61	
195	140	I/O	MD62	
196	141	I/O	MD63	接地
197	273	OUT	GND	
198	142	OUT	MA23	
199	143	OUT	MA24	
200	144	OUT	RESETO	リセット
201	145	OUT	TCSOL	モータ制御
202	146	OUT	TCSIL	モータ制御
203	147	OUT	PD15 PA15	ICチップ
204	148	OUT	PD14 PA14	
205	149	OUT	PD13 PA13	
206	150	OUT	PD12 PA12	
207	276	IN	VDD	3.3V電源供給
208	277	IN	GND	接地
209	151	OUT	PD11 PA11	ICチップ
210	152	OUT	PD10 PA10	
211	153	OUT	PD 9 PA 9	
212	154	OUT	PD 8 PA 8	
213	278	IN	VDD	3.3V電源供給
214	279	IN	GND	接地
215	155	OUT	PD 7 PA 7	ICチップ
216	156	OUT	PD 6 PA 6	
217	157	OUT	PD 5 PA 5	
218	158	OUT	PD 4 PA 4	

【表7】

219	281	IN	GND	接地
220	159	OUT	PD 3 PA 3	K119データ ROMアドレス
221	160	OUT	PD 2 PA 2	
222	161	OUT	PD 1 PA 1	
223	162	OUT	PD 0 PA 0	
224	282	IN	VDD	3.3V電源供給
225	163	IN	VCC	5V電源供給
226	164	IN	C7	E1RSデータ
227	165	IN	C6	
228	166	IN	C5	
229	167	IN	C4	
230	283	IN	GND	接地
231	168	IN	C3	E1RSデータ
232	169	IN	C2	
233	170	IN	C1	
234	171	IN	C0	
235	286	IN	VDD	3.3V電源供給
236	172	OUT	PROMCS	BIOS PROM
237	173	IN	BGVS	K119垂直同期
238	174	IN	BGHS	K119水平同期
239	175	OUT	MSSEL	K119マスタ
240	176	IN	BGCLK	バスセルクロック
241	177			余裕
242	178	IN	VCC	5V電源供給
243	289	IN	GND	接地
244	179	OUT	SCLK	K119シリアル
245	180	I/O	SDAT	K119シリアル
246	181	OUT	SFRS	K119フレーム 同期
247	182			余裕

【表8】

チップピン#	パッケージピン#	タイプ	名称	説明
248	183	IN	SERIAL	シリアル入力
249	184	OUT	SERIAL	シリアル出力
250	185	IN	A43SDP	K119同期
251	292	IN	VDD	3.3V電源供給
252	293	IN	GND	接地
253	186	IN	A43SCI	K119CODEC
254	187	OUT	A43SDI	K119CODEC
255	188	IN	A43SDO	K119CODEC
256	189	IN	VCC	5V電源供給
257	190	IN	TDI	JTAGテストデータ 入力
258	297	IN	GND	接地
259	191	IN	TCK	JTAGテストクロ
260	192	OUT	TDO	JTAGテストデー
261	193	IN	TRSTL	JTAGテスト
262	194	IN	TMS	JTAGテスト
263	196	OUT	INTAL	PCI用インタラプ
264	196	IN	RSTL	PCIインタラプト
265	298	IN	VDD	3.3V電源供給
266	197	IN	PCICLK	PCIクロック
267	198	IN	GNTL	PCIバス提供
268	199	OUT	REQL	PCIバスリクエスト
269	299	IN	GND	接地
270	200			余裕

【0026】1.5 ファームウェア構造

1.5.1 概要

MSPは、ベクトル化されたDSPファームウェアライブラリー（ベクトル処理器により実行される）の非常に最適化された結合及びシステム管理機能（ARM7により実行される）を通して強力かつ開放的な応用環境を多く提供する。MSPは信号処理開発とホスト応用開発とを分離することによって、スケール可能な性能、費用効果的なマルチメディア&通信、便利な使用及び容易な取扱い等を提供する。また、応用開発と維持費用を減少

させる。

1.5.2 ファームウェア構造

MSPファームウェアシステム構造は、図7の図示のとおりである。陰影領域はMSPシステム要素を表し、残りの余白は内在するPC応用及び動作システムを表す。

1.5.2.1 MOSA（マルチメディア動作システムの構造）

MSPの実時間動作システムカーネルは“MOSA”といい、これはマイクロソフトの実時間カーネルMMOSAのサブセットである。MOSAは実時間的で、丈夫で

あり、マルチタスキングが可能であり、先買権のある動作システムであって、MSP上で具現されたマルチメディア応用のために活用される。これは下記のような主要機能を遂行する。

*ホストウインド95とウインドズNTとのインターフェーシング

*ホストから選択された応用ファームウェアのダウンロード

*ARM7及びベクトル処理器で遂行するためMSPタスクのスケジューリング

*メモリ&I/O装置を含むすべてのMSPシステム資源の管理

*MSPタスク間の通信の同期化

*MSP関連のインタラプト、例外及びステータス条件のリポーティングMOSAはARM7上で排他的に動作する。より詳細なものは、MMOSA実時間カーネル仕様を参照する。

1. 5. 2. 2 マルチメディアライブラリモジュール
マルチメディアライブラリモジュールは、データ圧縮、MPEGビデオ&オーディオ、音声コーディング及び合成、サウンドブラスター互換可能なオーディオ等のような機能を遂行するボード範囲のモジュールを提供する。各モジュールは、MSP環境で最適化され、マルチタスキング環境で遂行するように設計される。

【0027】1. 5. 3 テレコムライブラリ

1. 5. 3. 1 概要

適切なDSPファームウェアと共に、MSPはインタセプトされる音声応用を支援し、かかってくる電話呼出に応答し、ハードディスク上にメッセージを貯蔵するように使用され得る。また、システムスピーカは、半二重(half-duplex)スピーカフォンをサービスするためにマイクロフォンを使用することができる。かかってくる電話及びかける電話の進行が感知され、システムで使

【0028】1. 6 プログラミングモデル

1. 6. 1 概要

ハードウェアの観点から見る時、MSPは2個のCPUと、多数の集積された周辺装置を含む単一チップの解決案である。ソフトウェアの観点から見る時MSPはPCIバス上に存在する高性能デジタル信号処理(DSP)装置である。ホストCPUによるMSPの制御は、下記のいずれか1つによって実現され得る。

*PCIバスを通してMSP制御&ステータスレジスタの読み取り/書込または

*ホストシステムメモリに存在する共有データ構造 *

*MSPローカルメモリに存在する共有データ構造

MSPプログラムの遂行は常にARM7 CPUから始まり、これは順次的にベクトル処理器にある第1従属的な実行ストリームを初期化させ得る。ARM7 CPUとベクトル処理器との間の制御同期化は、ARM7の任意のコプロセッサ命令(STARTVP、INTVP、TESTVP)と、ベクトル処理器における特殊な命令(VJOIN、VINT)によって遂行される。ARM7 CPUとベクトル処理器との間のデータ伝送はARM7で実行されるデータ移動命令によって遂行され得る。ARM7 CPUは一般的に、大部分のインタラプト&例外処理だけでなく、ホストインターフェース、資源管理、I/O装置処理を担当する。ベクトル処理器はすべてのデジタル信号処理及び、コプロセッサインタラプト(Arm7でベクトル処理器により発生される)とハードウェアスタックオーバーフロー(ベクトル処理器で)のような任意の特殊なインタラプトを担当する。また、MSPは多様なI/O装置に対してインターフェーシングするために集積された周辺機器を多く含む。すべての周辺装置のアドレスはメモリマッピングされ、よって標準メモリロード&貯蔵命令(Arm7 CPUまたはベクトル処理器の中のいずれか1つにより)でアクセスされ得る。

【0029】1. 6. 2 電源印加、リセット&初期化
電源が印加された後、MSPは機能を正確に確認するために、自動的にセルフテストシーケンスに入る。セルフテストシーケンスは下記のことを含む。

*すべての内部MSPレジスタの初期化

*MSPのすべての要素を確認するために、半導体チップのセルフテスト診断遂行

そして、セルフテストシーケンスは、<tds>秒近くまで持続されると予想される。セルフテストシーケンスの最後で、MSPは下記のものを含むMSPファームウェアを遂行する準備をする。

*MSPの初期化ソフトウェアのローディング及び実行

*MSPの実時間動作システム過0カーネルMMOSAのローディング及び実行

MSPは下記の3種類のリセットを支援する。

*PCIバスによるハードウェア制御システムリセット

*MSP制御レジスタにあるPCIシステムリセットビットによるソフトウェア制御システムリセット

*MSP制御レジスタにあるベクトル再開始、ビットによるソフトウェア制御再開始(restart)

【0030】1. 6. 3 PCI配列レジスタ

PCIバスに対するI/O装置であって、MSPはPCI Rev 2. 1に定義され、表9に示されているような一セットの構成レジスタを含む。

【0031】PCI配列レジスタ

【表9】

PCI 配列レジスタ

オフセット	説 明
0x00	装置及びベンダー確認子
0x04	状態及び命令語
0x08	クラスコード及び校正確認子
0x0C	多様なレジスタ
0x10	MSPベースアドレスレジスタ (MSP BASE)
0x14	仮想フレームバッファベースアドレスレジスタ
0x18	ベースアドレスレジスタ2 (非使用)
0x1C	ベースアドレスレジスタ3 (非使用)
0x20	ベースアドレスレジスタ4 (非使用)
0x24	ベースアドレスレジスタ5 (非使用)
0x28	予約
0x2C	予約
0x30	拡張ROMベースアドレス
0x3C	インタラプトライン

【0032】1. 6. 3. 1 装置&ベンダー識別子レジスタ

より詳細なものはPCIバス仕様Rev2. 1参照。

1. 6. 3. 2 ステータス&コマンドレジスタ

より詳細なものはPCIバス仕様Rev2. 1参照。

1. 6. 3. 3 クラスコード&校正識別子レジスタ

より詳細なものはPCIバス仕様Rev2. 1参照。
MSP-1EXに関してクラスコードは03に定義され、サブクラスは0である。

1. 6. 3. 4 その他のレジスタ

より詳細なものはPCIバス仕様Rev2. 1参照。

1. 6. 3. 5 MSPベースアドレスレジスタ (MSP BASE)

このレジスタはMSP装置のためのベースアドレスを貯蔵する。このアドレスはホストシステムソフトウェア(Windows 95/NT)により記入され、MSPハードウェアで使用されメモリをアドレッシングする。

1. 6. 3. 6 VFBベースアドレスレジスタ

このレジスタはVGA仮想フレームバッファのためのベースアドレスを貯蔵する。このアドレスはホストシステムソフトウェア(Windows 95/NT)により記入され、MSPハードウェアで使用されVGAフレームバッファをエミュレーションする。

【0033】1. 6. 3. 7 拡張ROMベースアドレス

より詳細なものはPCIバス仕様Rev2. 1参照。

1. 6. 3. 8 インタラプトラインレジスタ

より詳細なものはPCIバス仕様Rev2. 1参照。

1. 6. 4 ARM7 CPU

ARM7 RISC CPUはMSPのマスタ処理器であって、32ビットデータ経路を含んでおり、標準ARM7命令セット構造からなる。またARM7はベクトル処理器とインターフェースするために、特殊なコプロセッサ命令を含む。

1. 6. 5 ベクトル処理器

ベクトル処理器は、MSPのDSPエンジンであり、288ビットデータ経路を含んでおり、ARM7に対しコプロセッサとして動作する。このような機能はベクトル

処理器構造文書に記述されている。ベクトル処理器220は80MHzで動作し、6ステージのパイプラインすなわち、フェッチ(fetch)、デコード(decode)、流出人(issuer)、レジスタアクセス(register access)、実行(execute)及び記入(write)を含む。これはDSP関連処理のために最適化される。

【0034】1. 6. 6 仮想メモリ管理

MSP-1EXは仮想メモリ管理を支援しない。

1. 6. 7 インタラプト&実行処理

MSPでインタラプト&実行処理は大抵ARM7により行われる。内部のすべての入/出力装置インタラプトは、内部の8254インタラプトコントローラーに入って、これら間の優先順位を決め、最も高い優先順位のインタラプトを次の処理のためにARM7に送る。

1. 6. 8 物理的なメモリアドレスマップ

ARM7及びベクトル処理器プログラムは、図8に示したような物理的なメモリによってメモリマッピングされたすべてのMSP入/出力装置を示す。ARM7 (またはベクトル処理器) により示されるMSPアドレスマップは、0から始まって4GBまで拡張される。2GBから4GBまでの領域のアドレスは、次の関係式に従って、0から2GBまでのホスト(Pentium)PCIアドレスにマッピングされる。ホストPCIアドレス:=ARM7アドレス-8000 0000 (in hex) このようなマッピングによりARM7 (またはベクトル処理器) は、0から2GBまでのホストPCIメモリアドレスをアクセスするために、2GBから4GBまでのアドレスを使用することができる。ARM7は2GB以上のホストPCIメモリアドレスに対しては、アクセスできない。また、ホスト(Pentium)プログラムは、図9に示したような多少制限された物理的なメモリに従ってメモリマッピングされたすべての入/出力装置を示す。ホスト(Pentium)から見る時、
*MSP_BASEはMSPアドレスマップの始まりである。

*MSP_BASE+7DFFFFFFはMSPアドレスマップの最後である。

*MSPアドレスマップは、128MBの範囲のみで定

義される。

*【表10】

【0035】MSP I/O装置アドレスマップ

*

MSP I/O装置アドレスマップ

ARM7アドレス [31:0] hex	ホストアドレス [26:0] hex	装置
0000 0000	000 0000	内部ROM
0040 0000	040 0000	内部スクラッチパッドSRAM
0080 0000	080 0000	外部同期DRAM
0480 0000	480 0000	内部DRAMメモリコントローラー
0490 0000	490 0000	内部仮想フレームバッファコントローラー
04A0 0000	4A0 0000	内部DMAコントローラー
04B0 0000	4B0 0000	KS0119CODECシリアルライン
04C0 0000	4C0 0000	KS0122CODECシリアルライン
04C0 0200	4C0 0200	AD1843CODECシリアルライン
04D0 0000	4D0 0000	メモリデータ移動子
04D0 A000	4D0 A000	予約
07BF FFFF	7BF FFFF	予約
07C0 0000	7C0 0000	内部ビットストリーム処理器
07D0 0000	7D0 0000	内部8259インタラプトコントローラー
07D0 0010	7D0 0010	内部8254タイマー
07D0 0020	7D0 0020	内部16450UART
07E0 0000	7E0 0000	予約
07DF FFFF	7DF FFFF	予約
07DF FFF0	7DF FFF0	MSPホストコントロールレジスタ
07DF FFF4	7DF FFF4	MSP ARM7コントローラーレジスタ
07DF FFF8	7DF FFF8	予約
07DF FFFB	7DF FFFB	予約
07E0 0000		予約
7FFF FFFF		予約
8000 0000		他のホストPCI装置の0から7FFFまでマッピングされる。
FFFF FFFF		

【0036】1. 6. 9 MSPホスト制御レジスタ

※る特殊なレジスタを含む。

MSP-1EXは、ホスト（Pentium プロセッ

【0037】

サ）による初期化及び、インタラプトのために使用され※

【表11】

MSP制御レジスタ定義

ビット#	説 明
0	PCIシステムリセット
1	ARM7ベクトル再スタート
2	ホストからMSPインタラプトリクエスト(PENTIUM)
3	MSPからPCIホストインタラプトリクエスト
4	PCIホストインタラプト肯定応答
31:5	予約

【0038】bit<0> PCIシステムリセット。このビットはMSP関連のすべての内部/外部入出力装置を含む全体MSPシステムハードウェアを完全にリセットさせるために、ホスト（PENTIUM）で使用される。PCIシステムをリセットさせた後、MSPはARM7、ベクトル処理器及びI/O装置に対するチップ上のすべてのセルフテスト診断実行を含む、標

準リセットシーケンスを処理する。このようなリセットは、ハードウェアシステムリセットと同一な影響を及ぼす。

bit<1> ARM7&ベクトル処理器の再開始。このビットはARM7とベクトル処理器を再開始させるために、ホスト（PENTIUM）で使用される。この再開始はMSPが正常的なリセットシーケンスを全

然処理せず、チップ上のセルフテスト診断を全く実行しないとの意味で、完全なPCIシステムリセットと区別される。このビットが設定されると、ARM7はアドレス0から実行を開始し、ベクトル処理器はアイドルモード(idle mode)に入る。この時、どのような内部または外部I/O装置も影響を受けない。

bit<2> ホスト(PENTIUM)からのMSPインタラプト要請。このビットはMSPを直接インタラプトするためにホスト(PENTIUM)で使用され、ARM7をインタラプトするために使用される内部8259プログラム可能なインタラプトコントローラ(PIC)の入力の中のいずれか1つに連結される。このビットは、ホスト(PENTIUM)により設定さ

MSP ARM7制御レジスタ定義

ビット#	説 明
0	MSPからPCIホストインタラプトリクエスト
31:1	予約

【0041】bit<0> MSPからのPCIホストインタラプト。このビットは、PCIバス上のPCI INTA#ピンのアクティブ確認を通しホストをインタラプトするためにMSPで使用される。このビットはARM7により設定され、PCIバスを通しホスト(PENTIUM)によりクリアーされる。

bit<31:1> 予約

1. 6. 11 MSP内部μROM

内部ROMは全体16KByteからなり、次のことを※30

内部μROMアドレスマップ

アドレス (HEX)	説 明
	ARM7リセット及び初期化アドレス
	ベクトル処理器リセット及び初期化アドレス
	FAUマクロセル セルフテスト診断
	SRAMマクロセル セルフテスト診断
	ベクトル処理器コア セルフテスト診断
	キャッシュ制御器 セルフテスト診断
	メモリ制御器 セルフテスト診断
	PCIバス セルフテスト診断
	ビットストリーム処理器 セルフテスト診断
	DMA制御器 セルフテスト診断
	8254インターバルタイマー セルフテスト診断
	8259インタラプト制御器 セルフテスト診断
	16450UARTレジスタ セルフテスト診断
	KS0122シリアルライン セルフテスト診断
	KS0119シリアルライン セルフテスト診断
	AD1843 シリアルライン セルフテスト診断
	ARM7入出力装置インタラプトハンドラー0
	ARM7入出力装置インタラプトハンドラー1
	ARM7入出力装置インタラプトハンドラー2

【0043】

*れ、ARM7によりクリアーされる。

bit<3> PCIホストインタラプト認知。このビットはMSPが発生したPCIホストインタラプト要請を認知するために、ホスト(PENTIUM)で使用される。このビットはホスト(PENTIUM)により設定され、ARM7によりクリアーされる。

bit<31:4> 予約

【0039】1. 6. 10 MSP ARM7制御レジスタ

MSP-1EXは、ARM7プロセッサによりホストをインタラプトすることに使用される特殊なレジスタを有する。

【0040】

※含む。

*μROM初期化ソフトウェア

*セルフテスト診断ソフトウェア

*多様なシステム管理ソフトウェア

*多様なライブラリーサブルーチン

*命令及びデータ常数のためのキャッシュ

アドレスマップは、次の表に示したとおりである。

【0042】内部μROMアドレスマップ

【表12】

【表13】

ARM7入出力装置インタラプトハンドラー3
ハンドラーを除いたARM7命令
(ARM7による)ベクトル処理器インタラプト
ベクトル処理器区切り点例外
ベクトル処理器DSPライブラリーサブルーチン
MPEG-1ビデオライブラリーサブルーチン
MPEG-1オーディオライブラリーサブルーチン
MPEG-2ビデオライブラリーサブルーチン
MPEG-2オーディオライブラリーサブルーチン
サウンドブラスターライブラリーサブルーチン
V.34DSPライブラリーサブルーチン
テレコム利用ライブラリーサブルーチン
2Dグラフィックライブラリーサブルーチン
3Dライブラリーサブルーチン
H.261ライブラリーサブルーチン
H.263ライブラリーサブルーチン
G.728ライブラリーサブルーチン
G.723ライブラリーサブルーチン
ARM7データ常数
ベクトル処理器データ常数

【0044】1. 6. 12 MSP内部SRAM

内部のSRAMはMSPのベクトル&制御&ステータスレジスタ(VCSR)により決まれる選択事項によって、キャッシュまたはローカルメモリの機能を遂行する。

ローカルメモリモードにおいて、アドレス空間は位置<MCP_BASE>:040 0000から始まって、内部SRAM部にマッピングされる。

1. 6. 13 MSP内部の周辺装置

また、MSPは2個の内部バス、すなわち64ビット、80MHzで動作するFbusと、32ビット、40MHzで動作するIObus上に存在する多い周辺装置を有する。Fbus上の装置は次のことを含む。

- *外部の同期DRAMのためのメモリコントローラー
- *仮想フレームバッファインターフェース
- *外部PCIバスのためのPCIバスコントローラー
- *カスタマーASICインターフェース
- *8チャンネルDMAコントローラー

*メモリデータ移動器(ホストメモリとSDRAM間のデータ伝達のため)

*KS0122CODECシリアルライン

*KS0119CODECシリアルライン

20 *AD1843CODECシリアルライン

一方、IObus上の装置は下記のことを含む。

*8254-互換可能なプログラマブルインターバルタイマー

*8259-互換可能なプログラマブルインタラプトコントローラー(8レベル)

*16450-互換可能なUARTシリアルライン

*MPEGビットストリームデコーディング&エンコーディングのためのビットストリーム処理器

30 このような周辺装置等のレジスタアドレスマップは、表に示したとおりである。

【0045】

【表14】

*

内部周辺装置レジスタアドレスマップ

装 置	アドレス (HEX)	説 明
MSP DMA 制御器	4A0 0000	現在アドレスレジスタ0 (ビット<31:3>)
	4A0 0008	現在アドレスレジスタ1 (ビット<31:3>)
	4A0 0010	現在アドレスレジスタ2 (ビット<31:3>)
	4A0 0018	現在アドレスレジスタ3 (ビット<31:3>)
	4A0 0020	現在アドレスレジスタ4 (ビット<31:3>)
	4A0 0028	現在アドレスレジスタ5 (ビット<31:3>)
	4A0 0030	現在アドレスレジスタ6 (ビット<31:3>)
	4A0 0038	現在アドレスレジスタ7 (ビット<31:3>)
	4A0 0050	停止アドレスレジスタ0 (ビット<31:3>)
	4A0 0058	停止アドレスレジスタ1 (ビット<31:3>)
	4A0 0060	停止アドレスレジスタ2 (ビット<31:3>)
	4A0 0068	停止アドレスレジスタ3 (ビット<31:3>)
	4A0 0070	停止アドレスレジスタ4 (ビット<31:3>)
	4A0 0078	停止アドレスレジスタ5 (ビット<31:3>)
	4A0 0080	停止アドレスレジスタ6 (ビット<31:3>)
	4A0 0088	停止アドレスレジスタ7 (ビット<31:3>)
メモリー コントローラ	4A0 00A0	状態レジスタ
	4A0 00A8	制御レジスタA
	4A0 00B0	マスクレジスタ
VFB	4D0 0000	MSP現在アドレスレジスタ
	4D0 0008	ホスト現在アドレスレジスタ
	4D0 0010	MSP停止アドレスレジスタ
	4D0 0018	ホスト停止アドレスレジスタ
	4D0 0020	状態レジスタ
	4D0 0028	制御レジスタ

【表15】

装置	アドレス (HEX)	説明
KS ライオン	04C0 2000	フレームサイズレジスタ
	04C0 2001	ID
	04C0 2002	制御/DATAバイト
	04C0 2003	INDEX/DATA0
	04C0 2004	DATA1
	04C0 2005	DATA2
	04C0 2006	DATA3
	04C0 2007	予約
	04C0 2008	読出DATAシリアルインターフェース
	04C0 2009	予約
	04C0 200A	ロジック制御レジスタ
	04C0 200B	予約
	04C0 200C	予約
	04C0 200D	予約
	04C0 200E	状態レジスタ
	04C0 200F	予約
KS ライオン	04B0 0000	フレームサイズレジスタ
	04B0 0001	ID
	04B0 0002	制御/DATAバイト
	04B0 0003	INDEX/DATA0
	04B0 0004	DATA1
	04B0 0005	DATA2
	04B0 0006	DATA3
	04B0 0007	状態レジスタ
	04B0 0008	読出DATAシリアルインターフェース
	04B0 0009	読出PROMデータ
	04B0 000A	ロジック制御レジスタ
	04B0 000B	HS、VS極性
	04B0 000C	HSオフセット
	04B0 000D	VSオフセット
	04B0 000E	予約
	04B0 000F	予約
AD ライオン	04C0 4000 (記入のみ)	DAC1制御レジスタ記入データ入力
	04C0 5000 (記入のみ)	DAC2制御レジスタ記入データ入力
	04C0 6000 (記入のみ)	ADC左側制御レジスタ記入データ入力
	04C0 7000 (記入のみ)	ADC右側制御レジスタ記入データ入力
	04C04000+2	DAC1制御ワード入力
	04C05000+2	DAC2制御ワード入力
	04C06000+2	ADC左側制御ワード入力
	04C07000+2	ADC右側制御ワード入力
	04C0 4000 (記入のみ)	DAC1制御レジスタデータ出力
	04C0 5000 (記入のみ)	DAC2制御レジスタデータ出力
	04C0 6000 (記入のみ)	ADC左側制御レジスタデータ出力
	04C0 7000 (記入のみ)	ADC右側制御レジスタデータ出力

【表16】

装 置	アドレス オ(HEX)	説 明
A/D コン バー タ	04C06000+2	A/D 左側フラグレジスタ
	04C07000+2	A/D 右側フラグレジスタ
	04C0 6000	A/D 左側第1データ
	04C0 6000	A/D 左側第2データ
	04C0 6000	A/D 左側第3データ
	04C0 6000	A/D 左側第4データ
	04C0 7000	A/D 右側第1データ
	04C0 7000	A/D 右側第2データ
	04C0 7000	A/D 右側第3データ
	04C0 7000	A/D 右側第4データ
	04C0 4000	DAC 1 制御フラグレジスタ
	04C0 5000	DAC 2 制御フラグレジスタ
	04C0 6000	A/D 左側制御フラグレジスタ
	04C0 7000	A/D 右側制御フラグレジスタ
8259 コント ローラ	7D0 0000	初期化命令語ワード1
	未定義	初期化命令語ワード2 (非使用)
	未定義	初期化命令語ワード3 (非使用)
	7D0 0004	初期化命令語ワード4
	7D0 0004	動作制御ワード1
	7D0 0000	動作制御ワード2
	7D0 0000	動作制御ワード3
	7D0 0010	カウンタ#0レジスタ (R/W)
8254 タイマ	7D0 0014	カウンタ#1レジスタ (R/W)
	7D0 0018	カウンタ#2レジスタ (R/W)
	7D0 001C	制御ワードレジスタ (W)
	7D0 0020	受信レジスタ
1645 UART	7D0 0024	伝送レジスタ
	7D0 0028	除数レジスタ
	7D0 002C	インタラプト確認子レジスタ
	7D0 0030	インタラプトイネーブル除数レジスタ
	7D0 0034	ライン制御レジスタ
	7D0 0038	モデム制御レジスタ
	7D0 003C	スクラッチレジスタ

【表17】

装 置	アドレス (HEX)	説 明
ビデオ カウン タ	7C0_0000 7C0_0003	B P 処理モードレジスタ
	7C0_0004 7C0_0007	B P 制御レジスタ
	7C0_0008 7C0_000B	入力バッファ 0 のスタートアドレス
	7C0_000C 7C0_000F	入力バッファ 0 の終了アドレス
	7C0_0010 7C0_0013	入力バッファ 1 のスタートアドレス
	7C0_0014 7C0_0017	入力バッファ 1 の終了アドレス
	7C0_0018 7C0_001B	入力バッファ 0 のスタートアドレス
	7C0_001C 7C0_001F	入力バッファ 0 の終了アドレス
	7C0_0020 7C0_0023	入力バッファ 1 のスタートアドレス
	7C0_0024 7C0_0027	入力バッファ 1 の終了アドレス
	7C0_0028 7C0_002B	文脈貯蔵スタートアドレス
	7C0_002C 7C0_002F	入出力ダブルバッファで使用する最後データの バイトアドレス
	7C0_0030 7C0_0031	B P 状態レジスタの L S B
	7C0_0032 7C0_0033	B P 状態レジスタの M S B
	7C0_0050 7C0_0053	B P 状態レジスタ
	7C0_0054 7C0_0057	マクロブロック数において B P インタラプトマスク レジスタと画像垂直及び水平サイズ
	7C0_0058 7C0_0059	A R M 7 インタラプトリクエストピン
	7C0_005A 7C0_005F	未来拡張用領域
	7C0_0060 7C0_19F	B P キャッシュ領域
	7C0_01A0 7CF_FFFF	未来拡張用領域

【0046】内部周辺装置レジスタアドレスマップ

1. 6. 14 I O B U S 周辺装置

1. 6. 14. 1 8 2 5 4 - 互換可能なプログラマブル
インターバルタイマ

MSP は、下記のような機能を有するソフトウェアとして
使用するために、標準 8 2 5 4 - 互換可能なプログラ
マブルインターバルタイマを含む。

* 3 個の独立的な 1 6 ビットカウンタを有する。

* 6 個のプログラマブルカウンタモードを支援する。 *

* すべてのカウンタは、制御ワードレジスタに記入する
ものと初期カウントによりプログラムされる。

* 制御ワードレジスタ

このレジスタは、タイマに対する多様な制御情報を有
する。このレジスタのビット定義は、表に示したとおり
である。

【0047】

【表 18】

制御ワードレジスタ

ビット#	説 明
0	BCDカウンティングセレクト (省略時: バイナリ)
3:1	カウンティングモード: 000: モード0 001: モード1 x10: モード2 x11: モード3 100: モード4 101: モード5
5:4	記入/読出セレクト: 00: カウンタラッチ命令語 01: LSBのみ記入/読出 10: MSBのみ記入/読出 11: 最初にLSBを記入/読出してから次にMSBを記入/読出
7:6	セレクトカウンタ: 00: セレクトカウンタ0 01: セレクトカウンタ1 10: セレクトカウンタ2 11: セレクトカウンタ3

【0048】*ステータスレジスタ

このレジスタは、タイマーに対するステータス情報を有する。

*カウンタ0、1、2

この3個のレジスタは、主にタイマーによりカウンティングする素子である。各カウンタは16ビット幅を有し、プリセットが可能で、BCDモードの各2進数でカウントダウンする。このレジスタの入力、ゲート及び出力は、制御ワードレジスタに貯蔵されたMODESの選択により構成される。この3個のカウンタは完全に独立的である。

【0049】1. 6. 14. 2 8259-互換可能なプログラマブルインタラプトコント

ローラー (PIC)

MSPプログラマブルインタラプトコントローラーは、すべてのx86-基盤パーソナルコンピュータにおいて非常に一般的な標準8259であり、その機能は次のことを含む。

*8個レベルの優先順位を支援する。

*プログラマブルインタラプトモード

*個別的な要請マスク能力

MSP-1EXにおいて、8個レベルのインタラプト入力は、多様なI/O装置に対し下記のとおり割当てられ

る。

*レベル0 (最も高い) は8254タイマーに割当てられる。

*レベル1は、仮想フレームバッファ (VFB) に割当てられる。

*レベル2は、DMAコントローラーを含むカスタMASICロジックブロックに割当てられる。

*レベル3は、ビットストリーム処理器に割当てられる。

*レベル4は、PCIバスインターフェースに割当てられる。

*レベル5は<tbd>に割当てられる。

40 *レベル6は<tbd>に割当てられる。

*レベル7は、16550 UARTに割当てられる。インタラプトコントローラーの出力は、ARAM7 RISC CPUのインタラプト要請ライン (nFIQ) に結合される。

*レジスタ説明

ここには、下記のようなPICの動作を初期化することを使用される3個の8ビットレジスタがある。

*初期化コマンドワード1 (ICW1)

*初期化コマンドワード2 (ICW2): MSP-1EXには使用しない。

* 初期化コマンドワード3 (ICW3) : MSP-1 EXには使用しない。

* 初期化コマンドワード4 (ICW4)

また、下記のようなPIC動作を制御することに使用される3個の8ビットレジスタがある。

* 動作制御ワード1 (OCW1)

* 動作制御ワード2 (OCW2)

* 動作制御ワード3 (OCW3)

これらのすべてのレジスタは、アドレス部分 (bit < 0 >) とデータの部分の両方に特殊にエンコーディングされる。より詳細なことは、標準8259仕様を参照する。

[0050] 8259レジスタ説明

[表19]

8259レジスタ説明

名称	説明
ICW1<7:0>	初期化命令ワード1 ICW1は、MSP-1 EXの8259レジスタにのみ存在する。ICW1は、MSP-1 EXの8259レジスタにのみ存在する。ICW1は、MSP-1 EXの8259レジスタにのみ存在する。
ICW3<7:8>	初期化命令ワード3 (非使用)
ICW4<7:0>	初期化命令ワード4 (非使用)
OCW1<7:0>	動作制御ワード1 OCW1は、MSP-1 EXの8259レジスタにのみ存在する。OCW1は、MSP-1 EXの8259レジスタにのみ存在する。OCW1は、MSP-1 EXの8259レジスタにのみ存在する。
OCW2<7:0>	動作制御ワード2 OCW2は、MSP-1 EXの8259レジスタにのみ存在する。OCW2は、MSP-1 EXの8259レジスタにのみ存在する。OCW2は、MSP-1 EXの8259レジスタにのみ存在する。
OCW3<7:0>	動作制御ワード3 OCW3は、MSP-1 EXの8259レジスタにのみ存在する。OCW3は、MSP-1 EXの8259レジスタにのみ存在する。OCW3は、MSP-1 EXの8259レジスタにのみ存在する。

[0051] 1. 6. 14. 3 16450-互換可能なUARTシリアルライン

MSPは、外部シリアルI/O装置とのインターフェースとして使用される16450-互換可能なUARTシリアルラインを含む。より詳細なことは、標準16450仕様を参照する。

1. 6. 14. 4 ビットストリーム処理器

ビットストリーム処理器は、ビデオビットストリームデータを処理する特殊化されたロジックブロックであり、この機能は下記のことを含む。

* 可変長さハフマンデコーディング及びエンコーディング

* ジグザグ貯蔵フォーマットのビデオデータのアンパッキング及びパッキング

* 多様なビットレベル処理

ビットストリーム処理器は、同時的な処理ユニットとして動作し、ベクトル処理器またはARM7によりソフトウェアで制御される。より詳細なことは、ビットストリーム処理器部分を参照する。

1. 6. 15 FBUS周辺装置

FBUS周辺装置は下記のとおりである。

* カスタマASICロジックインターフェース

* 8個チャンネルDMAコントローラー

30 * 三星のKS0119に対するビデオエンコーダーシリアルラインインターフェース

* アナログデバイス社のAD1843に対するオーディオ&テレコムシリアルラインインターフェース

1. 6. 15. 1 ASICインターフェースロジックインターフェース

この節は、外部のすべてのCODECと、カスタマが規定したASICロジックブロックに対するインターフェースロジックを含む。このブロックのすべてはハードウェアで具現され、プログラマー可視(program-visible)レジスタは備えない。より詳細なことはASICインターフェース部分を参照する。

1. 6. 15. 2 DMAコントローラー

MSP-1 EXは、下記のような機能を有するチップ上のDMAコントローラーを備える。

* 8個の独立的なDMAチャンネル

* 個別的なDMAチャンネルに対するイネーブル/ディスエーブル制御

* メモリトランスファーまたは逆トランスファーに対するI/O装置

50 * アドレス増加及び減少

より詳細なことは、ASICインターフェース部分を参照する。

【0052】1. 6. 15. 3 メモリデータ移動器

また、MSP-1EXは、特殊なメモリデータ移動器を備える。このメモリデータ移動器は、ホスト（PENTIUM）メモリと、MSPローカルSDRAMメモリ間でデータを移動させるために使用される。メモリデータ移動器は、基本的に下記のようなレジスタを含む特殊なDMAコントローラーである。

*MSP現在アドレスレジスタ：この32ビットレジスタは、メモリデータ転送の初期にSDRAMメモリアドレスを定義する。このレジスタはARM7により記入または読出でき、初期値はARM7によりロードされなければならない。アドレスはデータ転送サイズに基づいて増加される。

*ホスト現在アドレスレジスタ：この32ビットレジスタは、メモリデータ転送の初期にホストメモリアドレスを定義する。このレジスタはARM7により記入または読出でき、初期値はARM7によりロードされなければならない。アドレスはデータ転送サイズに基づいて増加される。

*MSP停止アドレスレジスタ：この32ビットレジスタは、メモリデータ転送の最後にSDRAMメモリアドレスを定義する。このレジスタはARM7により記入または読出でき、MSP現在アドレスレジスタと比較し使用される。もし、これらがマッチングすると、メモリデータ移動器はMSPのEnd-Of-Process信号を発生する。

*ホスト停止アドレスレジスタ：この32ビットレジスタは、メモリデータ転送の最後に、ホストメモリアドレスを定義する。このレジスタはARM7により記入または読出でき、ホスト現在アドレスレジスタと比較し使用される。もし、これらがマッチングすると、メモリデータ移動器はホストのEnd-Of-Process信号を発生する。

*ステータスレジスタ：このレジスタは、メモリデータ移動器と関連したステータス情報を含む。ビットエンコーディングは下記のとおりである。

<0>：MSP EOP。このビットは、メモリデータ移動器がMSPの停止アドレスに到達したか否かを決定する。もし、ARM7がソース現在アドレスレジスタを初期化すると、ARM7は0080 0000(hex)に*

*セットされる。このビットはARM7により読出のみが遂行され、記入は遂行されてはいけない。

<1>：HOST EOP。このビットは、メモリデータ移動器がホストの停止アドレスに到達したか否かを決定する。もし、ARM7がホスト現在アドレスレジスタを初期化すると、ARM7は8000 0000(hex)にリセットされる。このビットはARM7により読出のみが遂行され、記入は遂行されてはいけない。

*制御レジスタ：このレジスタは、メモリデータ移動器と関連した情報を含む。このビットエンコーディングは下記のとおりである。

<0>：方向。このビットはデータ転送の方向を決定する。このビットが“0”（デフォルト）の場合、データ転送の方向はホスト（PENTIUM）メモリからMSP SDRAMメモリであり、このビット“1”の場合、データ転送の方向は、SDRAMからホストメモリである。このビットはARM7により記入されなければならない。

<1>：インタラプトイネーブル。このビットはメモリデータ移動器が、データ転送の最後にARM7をインタラプトするか否かを決定する。このビットはARM7により記入されなければならない。

<2>：DMAイネーブル。このビットはメモリデータ移動器が動作するようにイネーブルさせる。このビットはARM7により記入されなければならない。

<3>：データ転送サイズ。このビットが“0”（省略時）の場合、各メモリのデータ転送サイズは32バイトであり、“1”の場合は64バイトである。このビットはARM7により記入されなければならない。

【0053】1. 6. 15. 4 KS0119ビデオエンコーダーシリアルラインインターフェース

KS0119ビデオエンコーダーシリアルラインインターフェースは、下記のことを含む。

*コーデックからの読出データを含むダブルバッファ受信データバッファレジスタ

*コーデックへの記入データを含むダブルバッファ伝送データバッファレジスタ

*シリアルラインに対する多様な制御&ステータス情報を含む制御&ステータスレジスタ

【0054】

【表20】

KS0119ビデオエンコーダーシリアルラインインターフェースレジスタ

アドレス オフセット (HEX)	ビット幅	説明
00	8	受信データバッファレジスタ (RBUF)
01	8	伝送データバッファレジスタ (TBUF)
02	8	制御及び状態レジスタ (CSR)

【0055】制御&ステータスレジスタのビットエンコーディングは下記のとおりである。

bit<0>: 受信データのフル状態である。このビットはシリアルラインが、KS0119CODECから8ビットのデータを受信した場合に設定される。もしインタラプトイネーブル (bit<7>) が設定されると、インタラプト要請もARM7に発生される。

bit<1>: 伝送データバッファが空いている状態である。このビットはシリアルラインがKS0119にデータを送るよう準備されている場合に設定される。もしインタラプトイネーブル (bit<7>) が設定されると、インタラプト要請もARM7に発生される。

bit<7>: インタラプトイネーブル。このビットはARM7にインタラプト要請をイネーブルさせるために使用される。

1. 6. 15. 5 AD1843オーディオ&テレコムシリアルラインインターフェース

AD1843シリアルラインインターフェースは下記の30
ことを含む。

* コーデックから読出されたデータを含む一セットのダブルバッファリングされたレジスタ

* コーデックに記入しようとするデータを含む一セットのダブルバッファリングされたレジスタ *

* * シリアルラインに対する多様な制御&ステータス情報を含む制御&ステータスレジスタ

より詳細なことは、AD1843コーデックインターフェース部分を参照する。

1. 6. 16 命令性能

表21は、毎サイクルが12.5nsであるベクトル処理器サイクルカウントにおける命令性能を示す。外部メモリバス幅は64ビットで、40MHzのページモードクロックを有すると仮定する。すべての命令性能は、32バイトベクトルモードに与えられる。規則は下記のとおりである。

* ラス (ras) : 外部メモリが第1アクセスをすることに要求されるサイクル数。一般に75nsまたは6個のサイクルを必要とする。

* 待ち時間 (latency) : 第1命令を実行するためのサンプル数。

* レート (rate) : 類似した連続的な命令実行の間に存在するサイクル数。

待ち時間がレートと同一な場合、1つの数字のみが使用される。

【0056】命令実行性能

【表21】

命令実行性能

ニモニツク	(8/9ビット)	(16ビット)	(32ビット)	(32ビット)
VCACHE				
VLCB	待ち時間 = 4、レート = 2			
VLCB.off	待ち時間 = $ras + 9$ 、レート = ras			
VLR	待ち時間 = 3、レート = 1			
VLR.off	待ち時間 = $ras + 8$ 、レート = ras			
VL	待ち時間 = 3、レート = 1			
VL.off	待ち時間 = $ras + 8$ 、レート = ras			
VLD	待ち時間 = 4、レート = 2			
VLD.off	待ち時間 = $ras + 12$ 、レート = 8			
VLQ	待ち時間 = 6、レート = 4			
VLQ.off	待ち時間 = $ras + 20$ 、レート = 16			
VPFFCH	$ras + (\# \text{ of Cache lines}) \times 4$			
VLWS	待ち時間 = 4、レート = 1			
VLWS.off	待ち時間 = $ras + 8$ 、レート = ras			
VLI	1			
VSTCB	待ち時間 = 4、レート = 2			
VSTCB.off	待ち時間 = $ras + 9$ 、レート = ras			
VST	待ち時間 = 4、レート = 1			
VST.off	待ち時間 = $ras + 8$ 、レート = ras			
VSTD	待ち時間 = 5、レート = 2			
VSTD.off	待ち時間 = $ras + 12$ 、レート = 8			
VSTQ	待ち時間 = 7、レート = 4			
VSTQ.off	待ち時間 = $ras + 20$ 、レート = 16			
VSTR	待ち時間 = 4、レート = 1			
VSTR.off	待ち時間 = $ras + 7$ 、レート = ras			
VWBACK	$ras + (\# \text{ of Cache lines}) \times 4$			
CFC	定義されない	待ち時間 = 2 レート = 1	定義されない	
CTC				
MCR				
MCRS				
MRC				
MRCS				

【表22】

ニモニック	(8/9ビット)	(16ビット)	(32ビット)	(32ビット)
VMOV	待ち時間 = 1、レート = 1			
VCMOV				
VCMOVM				
VEXTRT				
VINSERT				
CPINT	定義されない	10	定義されない	
FORK		6		
RESUME		6		
SYNCH		4		
VCBR		2		
VCBRI		2		
VD1CBR		2		
VD2CBR		2		
VD3CBR		2		
VCJSR		2		
VCJSRI		2		
VCRSR		2		
VCINT		4		
VCJOIN		4		
VCCS		2		
VADD	1		2 (待ち時間)	
VADDH			1 (レート)	
VAND			定義されない	
VASL				
VASA				
VAAS3				
VASS3			2 (待ち時間) 1 (レート)	
VASUB				
VAVG				
VAVGH				
VAVGQ				
VCVTIF	定義されない	2 (待ち時間)	定義されない	
VCMPV	1		2 (待ち時間)	
VCNTLZ			定義されない	
VCVTB9				
VCVTFF	定義されない		2 (待ち時間)	

【表23】

ニモニツク	(8/9ビット)	(16ビット)	(32ビット)	(32ビット)		
VDIVI	1	1	1	定義されない		
VDIVS						
VSATU						
VEXTSGN2						
VESL						
VESR						
VEXTSGN3						
VSATL						
VASR						
VLSL						
VDIV2N	2 (待ち時間) 1 (レート)	2 (待ち時間) 1 (レート)	3 (待ち時間) 1 (レート)	3 (待ち時間) 1 (レート)		
VLSR						
VMAC						
VMACF						
VMACL						
VMAS						
VMASF						
VMASL						
VMAX						
VMUL					2 (待ち時間) 1 (レート)	2 (待ち時間) 1 (レート)
VMULA						
VMULF						
VMULL						
VNOR	1	1	1	定義されない		
VOR						
VROUND	定義されない		2 (待ち時間)	2 (待ち時間)		
VSHFL	1	1	1	定義されない		
VSHFLH						
VSHFLL						
VUNSHFL						
VUNSHFLH						
VUNSHFLL						
VSUB			2 (待ち時間) 1 (レート)	2 (待ち時間) 1 (レート)	2 (待ち時間) 1 (レート)	
VSUB2						
VSUBS						
VXNOR			1	1	1	定義されない
VXOR						
VXORALL						

【0057】第2章 DSPコア

本章は、ハードウェア及びソフトウェアデザイナーが示しているDSPコアの仕様に関して記述している。

2.1 概要

DSPコアは、MSPにおいて基礎的な要素であり、すべての演算に対して単独に責任を担う。このDSPコアは次のように構成される。

* 40 MHzで動作し、実時間OS、インタラプト及び例外処理、入出力装置管理等のような、汎用データ処理用として使用する32ビットARM7 RISC CPU。

* 80 MHzで動作し、離散余弦変換、FIRフィルタリング、くりこみ、ビデオのモーション推定等のようなデジタル信号処理用として使用されるベクトル処理器。このベクトル処理器はARM7により初期化され、ARM7と同時に動作可能で、特殊な制御命令によりARM7と同期される。

M7と同期される。

* 80 MHzで動作し、ARM7のための1KBの命令キャッシュと1KBのデータキャッシュ、ベクトル処理器のための1KBの命令キャッシュと4KBのデータキャッシュ、ARM7及びベクトル処理器のための共有の16KBの集積された命令&データキャッシュROMから構成されるキャッシュサブシステム。ベクトル処理器用のデータキャッシュは、ハードウェアまたはソフトウェアによって制御され得る。キャッシュサブシステムは、32ビットデータバスを通してARM7とインターフェースし、128ビットデータバスを通してベクトル処理器とインターフェースする。

* ビットストリーム処理器、インタラプトコントローラ、タイマー及びUARTのような多様な内部周辺機器等とインターフェースする32ビット、40 MHzの入力&出力バス (IOBUS)。

* PCIバスコントローラー、メモリコントローラー、DMAコントローラー及びカスタマASICロジックブロックとインターフェースする64ビット、80MHzの高速入/出力バス(FBUS)。DSPコアのブロック図は、図10の図示のとおりである。

【0058】2.2 ARM7 RISC CPU

2.2.1 概要

ARM7 RISC CPUは、汎用の32ビットRISCプロセッサコアである。このARM7 RISC CPUは、標準コプロセッサインターフェースを通しベクトル10 10
ル処理器とインターフェースし、実時間OS、IO装置インタラプト処理及びホストCPUとの通信のように、大部の非演算的な集中機能処理することに使用される。ARM7 CPUは下記のような特性を有する。

* 電力感性応用に理想的な極めて静的な動作。

* 低電力消費：0.6mA/MHz @3V製作。

* 高性能：25MIPS@40MHz (40MIPSピーク) @3V。

* 大小サイズのEndian動作モード

* 実時間応用のための高速インタラプト応答(40MHzで22クロックサイクル)

*

汎用レジスタ及びプログラムカウンタ

使用者	FIQ	監督者	アボート	IRQ	非定義
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
R8	R8_fiq	R8	R8	R8	R8
R9	R9_fiq	R9	R9	R9	R9
R10	R10_fiq	R10	R10	R10	R10
R11	R11_fiq	R11	R11	R11	R11
R12	R12_fiq	R12	R12	R12	R12
R13	R13_fiq	R13_abt	R13_irq	R13_irq	R13_und
R14	R14_fiq	R14_abt	R14_irq	R14_irq	R14_und
R15(PC)	R15(PC)	R15(PC)	R15(PC)	R15(PC)	R15(PC)

【0060】

【表25】

プログラムステータスレジスタ

使用者 モード	FIQ モード	監督者 モード	アボート モード	IRQ モード	非定義 モード
CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	SPSR_fiq	SPSR_svc	SPSR_abt	SPSR_irq	SPSR_und

【0061】2.2.3 例外

例外は、命令処理途中で発生する非正常な条件をい

* 簡単かつ強力な命令セット。

* 約6mm²の非常にコンパクトなレイアウト。

2.2.2 レジスタ

ARM7は31個の汎用レジスタと6個のステータスレジスタ、すなわち合計37個のレジスタを有する。プログラマーには、常に16個の汎用レジスタと1つ或いは2つのステータスレジスタが提供される。ユーザー、スーパーバイザー、IRQ、FIQ、Abort及びUndefinedのようなすべてのプロセッサモードで、R0とR15は直接にアクセス可能である。R15を除いたすべてのレジスタは汎用に使用され、データまたはアドレス値を維持させることに使用される。R15はプログラムカウンタ(PC)を維持する。ステータスはレジスタのCPSR-現在プログラムステータスレジスタは、ALUフラグと現在モードビットを有している。R14はサブルーチンリンクレジスタとして使用され、ブランチ及びリンク命令が遂行された時、1セットのR15データを受信する。他の場合は、R14は汎用レジスタとしても使用され得る。

【0059】汎用レジスタ及びプログラムカウンタ

【表24】

い、これは制御流れの変更を招来する。ARM7例外動作の7タイプに関し、上位優先順位から下位優先順位に列挙すると下記のとおりである。

*リセット(reset) (最上位優先順位)

*取消し(abort) (データ)

*FIQ

* * IRQ

*取消し(abort) (プリフェッチ)

*定義されていない命令トラップ、ソフトウェアインタラプト (最下位優先順位)

【0062】

【表26】

例外ベクトルテーブル

アドレス	例 外	エントリーモード
0000 0000	リセット	監督者
0000 0004	非定義命令トラップ	非定義
0000 0008	ソフトウェアインタラプト	監督者
0000 000C	アボート (プリフェッチ)	アボート
0000 0010	アボート (データ)	アボート
0000 0014	ARM7により貯蔵される	--
0000 0018	IRQ	IRQ
0000 001C	FIQ	FIQ

【0063】2. 2. 4 命令セット

すべてのARM7命令は条件的に実行されるが、これはARM7命令がCPSRレジスタにあるN、Z、C、Vフラグ値によって実行されるかもしくは実行されないことを意味する。ARM7命令は、下記のような多様なカテゴリーに分けられる。

*ブランチ及びリンクされたブランチ (B、BL)

*データプロセッシング (AND、EOR、SUB、RSB、ADD、ADC、SBC、RSC、TST、TEQ、CMP、CMN、ORR、MOV、BIC、MVN)

*PSRトランスファー (MRS、MSR)

*掛け算及び掛け算-累算 (MUL、MLA)

*シングルデータトランスファー (LDR、STR)

*ブロックデータトランスファー (LDM、STM)

*シングルデータスワップ (SWP)

*ソフトウェアインタラプト (SWI)

*コプロセッサデータ動作 (CDP) (これは一グループの命令である。)

*コプロセッサデータトランスファー (LDC、STC)

*コプロセッサレジスタトランスファー (MRC、MCR)

2. 3 ベクトル処理器

2. 3. 1 概要

ベクトル処理器は、最大性能のために、単位命令多重データ (SIMD) 構造を利用する強力なデジタル信号処理器であって、非常に優れた性能を実現させるために、多重データ要素上で並列に動作するパイプラインされたRISCエンジンから構成される。多重データ要素は、576ビットベクトルでバッキングされ、これは下記のようなレートで計算され得る。

*12.5ns-サイクル毎に32個の8/9ビット固定小数点算術演算または

*12.5ns-サイクル毎に16個の16ビット固定小数点算術演算または

*12.5ns-サイクル毎に8個の32ビット固定小数点または浮動小数点算術演算

2. 3. 2 実行パイプライン

ベクトル処理器は、命令を実行させるために図11の図示のとおり、6段階のパイプラインを利用する。大部の32ビットスカラー演算が、サイクル当り1つの命令比率でパイプラインされる一方、大部の576ビットベクトル演算は、2個のサイクル毎に1つの命令比率でパイプラインされる。すべてのロード&貯蔵(Loads&Stores)は算術演算と重なり、別途でロード&貯蔵ハードウェアにより独立的に実行される。設計の複雑度と性能を調和させるために、ベクトル処理器は資源及びデータ従属

30

40

50

性をチェックするためのハードウェアインターロックを順序とは関係なく使用し、命令等を発生するか実行することができる。この特徴は、ロード及び貯蔵によって、データキャッシュが紛失される期間の性能を、特に大幅改善する。

【0064】2. 3. 3 ハードウェアマイクロ構造
ベクトル処理器は、図12の説明のとおり、4個の主機能ブロックから構成される。

*命令語取出ユニット (IFU)

*命令語デコーダー&発行器

*命令語実行データ経路

*ロード&貯蔵ユニット (LSU)

命令語取出ユニットは、命令語の先取り (prefetch) 及び、ブランチとジャンプのような命令語のサブルーチンに対する流れを制御するプロセッシングを担当する。IFUは現在実行ストリームに対してプリフェッチされた命令語からなる16個のエントリキューと、ブランチターゲットストリームに対してプリフェッチされた命令語からなる8個のエントリキューを有する。IFUはサイクルごとに命令語キャッシュから8個の命令語が受信できる。命令語デコーダー&発行器は、すべての命令語に対するデコーディング及びスケジューリングを担当する。たとえ発行器は、実行資源とオペランドデータ有効性によって、非順次的な大部の命令語のスケジュールが可能であるが、デコーダーはサイクル当たり1つの命令語を処理することができ、常にIFUから順次的に到着する命令語を処理することができる。ベクトル処理器は12.5ns/cycleで動作する多数個の288ビットデータ経路 (図13参照) を通してその性能の大部分を実現し、この場合次のことを含む。

*サイクル当たり2個の読出及び2個の記入を支援することができる4個ポートを有するレジスタファイル

*8回の32ビット掛け算 (整数また浮動小数点フォーマット)、16回の16ビット掛け算及び32回の8ビット掛け算の中のいずれか1つの演算時ごとに、12.5nsを生成する8個の32×32並列掛け算器

*8回の36ビットALU演算 (整数または浮動小数点フォーマット)、16回の16ビットALU演算または32回の8ビットALU演算の中のいずれか1つの演算時ごとに、12.5nsを生成する8個の36ビットALU

ロード&貯蔵ユニットは、それぞれ図14の説明のように、288ビット幅を有する別個の読出&記入データバスを通して、データキャッシュとインターフェースするために設計されたものである。

【0065】2. 3. 4 インタラプト&例外

ベクトル処理器は、次の2つの特殊条件のみを認識する。

*ARM7プログラムによって実行されるCPINT (コプロセッサインタラプト) 命令語

*ベクトル処理器プログラムによって実行される、サブルーチン命令語へのネストされたジャンプ (nested jump) & 掛け算の結果のハードウェアスタックオーバーフロー
ベクトル処理器がこれらの2個の特殊条件を処理する、より詳細な方法に対しては、ベクトル処理器構造文書を参照すること。MCPから発生されるその他のインタラプト及び例外条件は、ARM7のみによって処理される。

【0066】2. 4 キャッシュサブシステム

2. 4. 1 概要

キャッシュ制御ユニット (CCU) はARM7コア、ベクトル実行ユニット (LSU、IFU)、メモリ (MCU、PCI、DMA、CODEC) 及びIOデバイス (BP、UART、タイマー、インタラプトコントローラ) とインターフェースする。CCUは高速 (80MHz) のFBUSと、低速 (20MHz) のIOBUSとインターフェースする。CCUは事実上、すべての内部CPUコアユニットと周辺IOデバイスと間の中央データ伝送ユニットである。MSPチップにおいてCCUの詳細な説明に関しては、MSP-1Eシステムスペックのブロック図 (pp. 1-10) を参照すること。非常に高性能のキャッシュサブシステムを支援するために、CCUの設計はすべての読出及び記入動作を支援するプロトコルに基づいたトランザクション (transaction) を使用する。メモリをアクセスする必要がある任意のユニットは、CCU制御ユニットでリクエスト (request) を発生させ得る。制御ユニットにあるアービタ (arbiter) は、固定された優先順位に基づいてリクエストを承認し、リクエスター (requestor) で 'transaction_id' を返信する。リクエスターはこの 'transaction_id' を貯蔵し、データが実際に到着した場合に、返信されたデータが認識できるようにする。CCU制御が1つのユニット (キャッシュミス (cache miss) が発生した場合、多いサイクルを必要とする場合もある) からのリクエストを処理する間、他のユニットから新しいリクエストが、他の 'transaction_id' と共に次のサイクルで承認される場合もある。リクエストをペンディング (pending) させるこのような方法では、他のユニットからの連続的なリクエストを遮断させることが発生されないで、高性能の実験が可能になる。現在、CCUは1つのサイクルで1つの読出リクエストと、1つの記入リクエストを同時にアクセプトし承認することができる。

【0067】メモリに対するインターフェースユニット (FBUS) は、4個エントリーのアドレスキューと、1個エントリーのライトバック (write-back) ラッチからなる。最善の状態では、FBUSはARM命令語キャッシュからの1つのペンディングリフィール (読出) リクエスト、VEC命令語キャッシュからの1つのペンディングリフィール (読出) リクエスト、VECデータキャッシュからの1つの記入リクエストと、ダーティ (dirty

y) キャッシュラインにより、VECデータキャッシュからの1つのライト-バックリクエストを支援することができる。また、キャッシュメモリ自体は、高性能のために最適化される。MSPキャッシュシステムは、チップ上(on-chip)のキャッシュSRAMとキャッシュROMとを有する。キャッシュSRAMは、ARM CPUとベクトルコアまたは命令語とデータ間のスラッシング(hrashing)を防止するため、4個の相互に異なるバンクからなる。キャッシュROMは、ARM7とベクトルコアのために高速及び高密度のデータ貯蔵領域を提供する。例えば、タグ(tag)がキャッシュROMに対して変更されることはないが、有効ビットの使用が不可能になり、データが外部メモリから返還される。要すれば、チップ上のキャッシュメモリは、次のようなブロックを含む。

* 1KBの直接マッピングされた命令語キャッシュと、1KBの直接マッピングされつつARM7に対する32ビットデータバスインターフェースを有するライト-バックデータキャッシュ

* 1KBの直接マッピングされ、ベクトル命令語フェッチユニットに対する256ビットバスインターフェースを有する命令語キャッシュ

* 4KBの直接マッピングされ、ベクトル実行ユニットに対する256ビットバスインターフェースを有するライト-バックデータキャッシュ。データキャッシュはデュアルポートからなり、80MHzのサイクルごとに256ビットの読出データを提供し、256ビットの記入データを支援することができる。

* 4KB VECデータキャッシュは、ソフトウェアの制御下で、スクラッチパッド(scratch-pad)演算により形成できる。

* ARM7及びベクトル処理器で使用するために共有または集積された命令語&データROMキャッシュ。ARM7に対するインターフェースは、その命令語キャッシュと同一な32ビットバスを通して、ベクトル処理器に対するインターフェースは、その命令語キャッシュと同一な256ビットを通してなる。

* 5個のポート：

- ARM7のための読出/記入ポート
- ベクトル処理器の命令語取出ユニットのための読出ポート
- ベクトル処理器のロード/貯蔵ユニットのための読出/記入ポート
- ベクトル処理器のIOBUSのための読出/記入ポート
- FBUSのための読出/記入ポート

* ARM7 CPU命令語キャッシュのための32×256ビットSRAM(～1KB)

* ARM7 CPUデータキャッシュのための32×256ビットSRAM(～1KB)

* ベクトル処理器データキャッシュのための128×56ビットSRAM(～4KB)

* ベクトル処理器命令語キャッシュのための32×256ビットSRAM(～1KB)

* データ&命令語キャッシュのための512×256ビットSRAM(～16KB)

ベクトルデータキャッシュの制御は、ハードウェア制御またはソフトウェア制御によって遂行される。

【0068】2.4.2 キャッシュサブシステム構造
図15は、MSPキャッシュシステムのブロック図であり、次のブロックIDC(Instruction Data Cache)、キャッシュROM、CCU_DATA_DP、CCU_ADR_DP、CCU_CTL及びCCU_SMとから構成される。それぞれのサブブロックは、さらに詳細なことは後述する。

2.4.2.2 IDC

命令語及びデータキャッシュ(IDC; 図16参照)は、チップ上のSRAMメモリであり、命令語及びデータキャッシュアクセスを提供するために使用される。このキャッシュは、1つのアレーに対し4個のバンク：ARM_IC(1KB)、ARM_DC(1KB)、VEC_IC(1KB)及びVEC_DC(4KB)から構成される。任意のサイクルで、このキャッシュは1つの読出リクエストと1つの記入リクエストをアクセプトする。タグRAMは、2個の読出ポートを有する。読出ポートアドレスと記入ポートアドレスとは、ヒットまたはミス条件に対し、内部キャッシュタグと比較される。データRAMは、読出ポートアドレスによりアクセスされる1つの読出ポートのみを有する。また、タグRAMとデータRAMとは、相互に異なるセットの記入アドレスを使用し記入される。従って、キャッシュアレーをアクセスするためには、4セットのキャッシュバンク選択信号と、3セットのラインインデックスを必要とする。IDCは下記のような特性を有する。

* ライト-バック規則に直接マッピングされる。

* キャッシュラインサイズは64Bであるが、データ幅は32Bであり、これはMSPチップのベクトルデータ幅のサイズに該当する。

* 各ラインは2個の有効ビットを有するが、1つはハイベクトルのためのものであり、他の1つはローベクトルのためのものである。また、データキャッシュはそれぞれのデータに対して1個づつすなわち、2個のダーティビットを有する。

* ARM_IC、ARM_DC及びVEC_ICのためのタグサイズは22ビット(アドレスビット10～ビット31)であり、VEC_DCのためのタグサイズは20ビット(アドレスビット12～ビット31)である。

* ARM_IC、ARM_DC及びVEC_ICのためのラインインデックスビットは5ビット(アドレスビット5～ビット9)であり、VEC_DCのためのライン

を切換する。また、ARM7に連結された全体的なりセット信号は、CLK1とMCLKがローの場合に解除(d e-assert)される。このような方法によって、2個のユニットは適切に同期化される。ARM7は命令語とデータ用として1つの入力バス (ARM_DATA<31:0>)のみを有するが、MSPチップは専用の命令語キャッシュ (ARM_IC、1KB) とデータキャッシュ (ARM_DC、1KB) とを備える。CCUは、ARM_NOPCを使用し、この二種類のリクエストを区別することができる。性能を更に向上させるために、CCUはメインキャッシュとARM7コアとの間に位置するマイクロ命令語キャッシュ (UI_CACHE、32B) とマイクロデータキャッシュ (UD_CACHE、32B) を付加する。このキャッシュは、それぞれ連続的なコードとデータとからなっている8ワードを有する。これらのマイクロキャッシュは、その自体のタグ (27ビット)、タグ比較器と有効ビットを有する。有効ビットはシステムリセット期間の間すべてがクリアされる。ARM7マイクロキャッシュは、実際のキャッシュよりはかえってプリフェッチバッファの役割を遂行する。ARM7読出の期間の間、アドレス (ARM_A、31:0) は常にタグに比較される。ヒットはARM_DATA<31:0>を通して命令語またはデータをリードバック(read back)する。その後1つのマイクロキャッシュはアドレス、データタイプ及び他の制御情報と共にリクエストをCCUに送る。CCUのアービタロジックは、すべてのユニットからのリクエストが読出リクエストを作ることを承認する。現在、承認を得ることにおいて、ARM7は他のブロックに対して最上位の優先順位を有する。その理由は、ARM7のマイクロキャッシュがミスを持たない限り、ARM7がリクエストを作る場合が殆どないからである。しかし、CCUは多数個のサイクルリクエストまたはアドレスキュー充足条件を提供するために、内部のホールドサイクルを有することができる。この期間の間、外部のリクエストは全然承認されない。ARM7からの記入は、アドレスがUD_TAGをヒットする場合、常にUD_CACHEを無効化させる。ライトスルー(write-through)またはライトバック(write-back)キャッシュとしてUD_CACHEを設計することにおいて、何等の試みも行っていない。UD_CACHE記入ヒット時に無効化させることにより、ARM_DCとUD_CACHE間のデータを一致させることができる。CCUはARM_ICまたはARM_DCに読出または記入リクエストを送る間にarm_nwaitを制御する。一般的に、CCUは記入期間の間には、arm_nwaitをホールドさせない。一応、記入リクエストがccu_write_hold2を見ないで承認されると、ARM7はただ次のサイクルからARM_DATA<31:0>にあるデータを持ってくる。CCUはデータを貯蔵するために、内部

の記入バッファを有する。ARM7は命令語を実行し続けることができる。しかし、CCUはたとえデータがメインキャッシュにあるとしても、常に1つのサイクルに対してarm_nwaitをホールドさせる。もし、読出リクエストがメインキャッシュをミスした場合、データが外部のメインメモリから返還されるまで、更に多いサイクルがホールドされる。図19に図示したARM_CCUインターフェース状態のマシンは、CCUがarm_nwaitを制御する条件を説明する。

【0072】図19において:

START: リクエストがなく、または読出データが返還されるか、ホールドせず記入リクエストが発生された場合の状態マシンのためのスタート状態。

HOLD: CCUは読出または記入のためのARM7リクエストを承認し、ホールド信号で承認を取り消す。

TAG: CCUは読出アドレスでタグをチェックする。

MISS: 読出アドレスは1つのミスを含み、ccuはリフィールリクエストを外部のDRAMに送る。

DATA: 読出データが返還され、CCUは返還されたデータをマイクロデータキャッシュへ送る。

2. 4. 3. 3 FBUSインターフェース

CCU_FBUSインターフェース状態マシン (F_SM) は、図20の図示のとおりである。図20において:

IDLE: アイドル状態

REQ: 読出または記入リクエストをFBUSアービタに送る。

GRT1: 承認サイズが8Bより大きい。

GRT2: 承認サイズが16Bより大きい。

GRT3: 承認サイズが24Bより大きい。

GRT4: 最後のサイクルに対する駆動データ

データ受信状態マシン (D_SM) は、図21の図示の通りである。図21において、

IDLE: アイドル状態

ONE: Fdata<63:0>から第1の8Bデータを受信する。

TWO: Fdata<63:0>から第2の8Bデータを受信する。

THREE: Fdata<63:0>から第3の8Bデータを受信する。

FOUR: Fdata<63:0>から第4の8Bデータを受信する。

REFILL: データをリクエストに返還する前、IDCをリフィールする。

RDY: データをリクエストに返還する準備をする。

【0073】2. 4. 4 読出及び記入動作

読出及び記入状態マシンは、図22の図示のとおりである。

2. 4. 4. 1 読出動作

MSPでIDC(Instruction and Data Cache)は3段のパイプラインサイクル: リクエストサイクル、タグサイクル及びデータサイクルで動作する。キャッシュヒット状況で、IDCは毎サイクルで命令語またはデータの返還が可能である。キャッシュコントローラユニット

(CCU)は、キャッシュSRAMアクセスのためにARM7、ベクトル処理器ユニット、FBUSとIOBUS間の仲裁を担当する。CCUはこの4個のマスタからのバスリクエストを監視し、特定のID番号を有する勝者10にバスを承認する。CCUはまたキャッシュをアクセスしタグを比較するために、キャッシュアドレスバスと読出/記入制御信号を発生する。キャッシュヒットがある場合、仲裁から勝ったバスマスタは、読出/記入動作のためにキャッシュをアクセスすることができる。キャッシュミスがある場合、CCUはメインメモリから返還される紛失データを待たずリクエストを発生させてから、バスマスタを助けてやる。それで、キャッシュミスを有するバスマスタは、ID番号を維持すべきである。以降、リクエストされたデータがキャッシュにあると、20CCUはGRANT信号を同一なID番号を有するデータを紛失したバスマスタに送る。このバスマスタはデータをアクセプトするかまたは無視する。キャッシュミスが発生した場合、メインメモリからデータを受けるために、ラインフェッチが遂行される。ラインサイズは64バイトに定義され、従ってCCUはメインメモリからキャッシュにデータを供給するために、8回の連続的なメモリアクセス(毎回64ビット)を実行する。

*リクエストサイクル: CCUはCLK1で多数個のユニット(ARM、IFU、LSU、IO)から読出リクエストをアクセプトする。リクエストは、CLK1の初期に、リクエスト信号(isu_req)と読出/記入信号(isu_rw)を表示する。CLK1の終わりでCCUはccu__grant__id[9:0]を駆動することによって、この読出リクエスト中の1つを承認する。ccu__grant__id[9:6]がリクエストのunit__idと整合されると、リクエストが承認される。リクエストはccu__grant__id[5:0]が、リクエストに関連したtransaction__idであるので、ccu__grant__id[5:0]をラッチしなければならない。リクエストが承認されると、リクエストはアドレス(isu_adr[31:0])とCLK2でキャッシュオフ動作(ish_ccu_off)及びデータタイプ(isu_vec_type[1:0]、isu_data_type[2:0])のような他の制御情報をCCUに送る。CLK2の終わりでccu_rd_hold_2が表示されなければ、リクエストは完全にCCUに送られ、リクエストされたデータはしばらく後で返還される。しかし、ccu_rd_hold_2が表示されると、CLK1で承認されたリクエストは取り消しつつ、リクエストは続いてアドレスと制御情報を送る。以前のすべてのgr 50

ant_id情報がまだ有効であるので、次のサイクルでは同一な読出リクエストを更に発生させる必要がない。ccu_rd_hold_2はCLK2でCCUによって解除されるまで、CLK1で一定に維持される。ccu_rd_hold_2はタイミング臨界信号であって、リクエストでCCUが現在サイクルで他のことを処理することに忙しくて、承認されたリクエストはまだ処理されていないことを知らせることに使用される。

*タグサイクル

リクエストが承認され、後でリクエストサイクルで取り消されなかった場合、リクエストはキャッシュアクセスのタグ比較段階に入る。CCUはタグ読出のためのラインを選択するために、アドレス(isu_adr[11:5])とバンク選択信号(リクエスト)を使用する。タグヒット信号(ccu_isu_hit_2)は、CLK2の終端で知られる。データはヒット状況のために次のサイクルで復帰される。読出ポートタグが出力され、CLKによりラッチされる。また、アドレスキューステータスは、このサイクルで評価される。タグミスと'almost_full_address_queue'は、'ccu_rd_hold_2'信号を表示する。CCU状態マシンは、或る新しい読出リクエストも処理しないが、中止されたタグ比較を更に試みる。それぞれのキャッシュライン(64B)は2つのベクトルを含むので、タグヒットを得るためにアクセスされたベクトルの有効ビットが有効でなければならない。2倍のベクトル(64B)データの読出のためには、タグヒットを得るために、2つの有効ビットが有効でなければならない。cc_off動作は常にタグミスを誘発させ、リクエストはアドレスキューに揭示される。

*データサイクル

これはCCUがデータをリクエストに復帰するサイクルである。データはCLK1で駆動される下位16Bと、CLK2で駆動される上位16Bと共に、ccu_dout[127:0]上に乗せられる。64Bデータリクエストの場合、伝送を終結させるために、1つの付加的なサイクルが使用される。CCUはデータが次のCLK1で復帰されることをリクエストに知らせるために、常にccu_data_id[9:0]をCLK2の初期の1/2サイクルで駆動する。リクエストは適切な戻りデータのために、常にccu_data_id[9:0]を比較する。また、戻りデータの指示子としてタグヒットが使用される。もし、タグサイクルでタグミスがあり、アドレスキューが充満でなければ、CCUはCLK1で紛失されたアドレス、id情報及び他の制御情報を、4個エントリーアドレスキューに揭示しつつ、キャッシュラインフェッチを始める。現在、それぞれのアドレスキューは、大略69ビットの情報を含む。CLK2でメモリアドレスラッチがロードされ、FBUSリクエストが次のCLK1で発生される。

【0074】2. 4. 4. 2 記入動作

IDCで記入動作は、3段のパイプラインサイクル: リ

クエストサイクル、タグサイクル及びデータ記入サイクルで動作する。記入アドレスヒット状況で、IDCは毎サイクルでキャッシュデータアレーにデータを記入することができる。

*リクエストサイクル: CCUはCLK1で多数個のユニット(ARM、LSU、IO)から記入リクエストをアクセプトする。リクエストはCLK1の初期にリクエスト信号(lsu_req)、読出/記入信号(lsu_rw)とベクトルタイプ(lsu_vec_type[1:0])を表示する。CLK1の終わりでCCUは、この記入リクエストの中のいずれか1つを承認する。相互に異なるユニットに対する記入承認は、承認信号(ccu_lsu_wr_grant)を直接リクエストユニットに表示することにより実現される。返還されるデータが全然ないので、リクエストユニットがCCUからtransaction_idを受信する必要はない。CLK2で、リクエストはアドレス(lsu_adr[31:0])、cc_off信号(lsu_ccu_off)及びデータタイプ(lsu_data_type[2:0])を供給しなければならない。読出の場合も同様に、たとえ、リクエストは承認されたが、現在サイクルで処理されなかったことをリクエストに知らせるために、CCUはCLK2の2の終端の近くでccu_wr_hold_2を表示する。リクエストはccu_wr_hold_2が解除されるまでアドレス、cc_off信号とデータタイプ情報を駆動し続ける。以降、次のサイクルでリクエストは記入データをccu_dout[127:0]に供給する。

*タグサイクル

リクエストが承認され、後でリクエストサイクルで取り消されなかった場合、リクエストはキャッシュアクセスのタグ比較段階に入る。このサイクルは記入ポートアドレスタグを比較する。CCUはキャッシュ用ラインを選択するために、アドレス(lsu_adr[11:5])とバンク選択信号(リクエスト)を使用する。タグヒット信号(ccu_lsu_hit_2)は、CLK2の終わりへ知られる。cc_off記入は、常にタグミス誘発させ、記入データは外部の記入のためにFBUS上に乗せられる。リクエストはCLK1の下位16BとCLK2の上位16Bにより、ccu_din[143:0]にデータを駆動し始める。64Bデータ転送の場合、データを駆動するためにリクエストは、1つの付加的なサイクルを取る。CCUはこのデータをホールドするために、内部の記入データラッチを有する。この記入がキャッシュをヒットさせるか(実際にデータをキャッシュに記入するために、1つまたは2つのサイクルが使用される)、キャッシュをミスさせる場合(データを記入するために、最も少ないサイクルが使用される)、リクエストは記入が完了されたことと見なす。

*データ記入サイクル

このサイクルは、キャッシュヒット状況のために、CCUが実際のデータをキャッシュに記入するサイクルである。タグサイクルでタグミスがある場合、CCIはこれをデータタイプによって相異なるように処理する。デー

タタイプが32Bで、ラインがクリーン(clean)の場合(2つのベクトルもクリーン)、CCUはただ現在のラインを、新しいタグと新しいデータをオーバーライトする。また、アクセス中のベクトルを有効及びダーティなもの并表示する反面、同一なラインの他のベクトルは無効なものに置いておく。データタイプが32Bより少ない場合、このサイクルは部分的にデータ記入が行われる。この部分データは、一時的なレジスタに貯蔵される。CCUは紛失された半ライン(32B)をメモリからフェッチしてからロードし、キャッシュに戻す。その後、部分データは適切なバイトイネーブル信号と共にキャッシュラインに記入される。ダーティキャッシュラインを有するすべての記入ミスに対して、CCUはまずダーティラインをコピーする。ダーティデータがまだ使用されていないので、CCUは承認ロジックにホールドを表示し、新しい読出または記入リクエストが承認されないようにする。その後、ダーティキャッシュラインデータをフェッチするために、ダーティラインを使用し内部の読出が始まる。結局、ライトバックアドレス及びデータはメモリに供給される。

【0075】2. 4. 5 プログラミングモデル

キャッシュサブシステムのすべては、ロード&貯蔵命令語を使用したハードウェアで制御されるので、ソフトウェア可視(software-visible)レジスタを必要としない。

2. 4. 6 IDC及びROMアドレスフォーマットは図23の図示のとおりである。

【0076】第3章 IOBUS説明

本章は、ハードウェアデザイナーが示すIOBUSの仕様に関して記述したものである。

3. 1 概要

IOBUSは、システムで使用される低速の標準的な周辺装置のために設計されたものである。このバスは、MSPキャッシュ制御ユニット(CCU)、ビットストリーム処理器(BSP)とタイマー/インタラプトコントローラと、UARTのようなすべての他のIO周辺装置等間のメインインターフェースの役割を果たす。バスのフォーマットは、インテル社のIOバスと非常に類似している。バスアービタ制御ロジックは、リクエストに対しバスを常に監視し、ラウンドロービン(round-robin)システムを用い、適切なリクエスト承認を発生させる。潜在的なバスマスタは、常にバスリクエストを表示し、バスを占有する前にバス承認が表示されることを待つ。バスマスタは、常にプロトコルによる期間の間、アドレスと制御ラインとを駆動する。

【0077】IOBUSは全体的に40MHzで動作する同期バスである。MSP IOBUS上でのすべての承認は、リクエストがアクティブにサンプリングされてから第1番目のサイクルで発生する。このバスは4個のサイクル(4個のバースト)に対し、16バイト伝送まで

処理可能である。これはバスマスタによりリクエストされた伝送サイズをバスアービタに知らせる2個のサイズビットを使用することによって実現される。IOBUSは32ビットアドレスとデータマルチプレクサーを有する。アドレスは常にデータの以前に現れる。IOB_ALE (アドレスラッチイネーブル) 信号はアドレスをラッチするために、受信装置により使用される。たとえ、8ビットデバイスがバスに連結されても、すべてのバスアクセシングは32ビット伝送を仮定する。正常的な規則によると、8ビットデバイスは、バスの下位8ビット [7:0] を使用し、16ビットデバイスはバスの下位16ビット [15:0] を使用する。もし16ビットデバイスが8ビットデバイスとの通信を願うと、8ビットデバイスがデータを探してラッチできるように、バスの下位8ビットに正確なデータを載置すべきである。同一期間に多数個のリクエストがある場合、承認されないリクエストは、IOBUSアービタが承認するまで、常にそのリクエストをホールドさせなければならない。このようなシステムにおいて許容されたリクエストに対し多い“バス-アクセシングサイクル”すなわち、4*32 20ビット伝送 (最大16バイト) がある。ブロック伝送は、常にそれぞれ多数個の32ビット伝送に分けられる。すべてのバス承認は、IOBUSアービタにより発生される。しかし、常にアドレス (有効時) を監視し、目的地に適切なチップ選択 (次のクロックサイクルに対*

*して) を発生させる並列デコーディングロジックがある。チップ選択は、常にただ1つサイクルに対して有効であり、以降アドレスがすべての読出及び記入リクエストのために表示される。それぞれのIOBUSノードは入力として専用のチップ選択を有する。ピン説明及びタイミング図を参照すること。2ビットサイズ情報は、バスアービタから承認されてからマスタによって発生され、以降2個のバスサイクルに対して有効である。CSがバス伝送サイクルを決定するために表示されると、選択されたスレーブはサイズ情報を獲得しなければならない。また、読出または記入時、IOBUSアービタは新しいリクエストを探し始める前、バスサイクルが終了されることを判断するための伝送サイズのトラックを維持する。バースト-バス伝送時 (読出または記入時) データ間には差異が全然ない。データ読出伝送において、データが有効な時点をリクエストに知らせ、このデータラッチを始めるためにREADY信号が使用される。このREADY信号は、バスマスタとスレーブにより発生される。このプロトコルを満足させるためには、すべてのIOBUSノードは、リクエストを処理する前、IOBUSインターフェースを設計する必要がある。このインターフェースは次のスペックを満足させなければならない。

【0078】

サイズ	説 明
2'b00	省略時。伝送サイズ=4バイト (1つの32ビットサイクル)
2'b01	伝送サイズ=8バイト (2つの32ビットサイクル)
2'b10	伝送サイズ=12バイト (3つの32ビットサイクル)
2'b11	伝送サイズ=16バイト (4サイクル用の最大サイズ)

【0079】3.2 ピン説明

以下、バスマスタ側から見たシステムIOBUSのためのアドレス、データ及び制御信号の定義を説明する。IOBUS構造定義を示している図24を参照すること。40
上述のごとく、IOBUSは多重化されたアドレス/データバスである。“xxx”はリクエスト名称(cc ※

※u, bsp, urt, tmr, int) を示す3個の文字コードである。

*システムIOBUS信号定義

【0080】

【表28】

*システムIOBUS信号定義

IOBus信号名	サイズ	方 向	コメント
IOBus[31:0]	32	両方向	32ビット両方向及びマルチプレクスされたアドレス及びデータバス
IOB_rd_l	1	入力/出力	アクチブロー。マスタ出力。スレーブ入力。
IOB_wr_l	1	入力/出力	アクチブロー。マスタ出力。スレーブ入力。
IOB_ready_l	1	入力/出力	アクチブロー。データが(スレーブモードで)読出有効時マスタを確認。マスタ入力、スレーブ出力。
IOB_cs xxx_l	1	入力	装置選択、アクチブロー。アドレスがラッチされてからリクエストを処理するために使用。
IOB_ale_l	1	入力	アクチブロー、アドレスが安定的な時を特定。アービタ出力、マスタ/スレーブ入力。
IOB_req xxx_l	1	出力	アクチブロー、IOBusを制御しマスタになるためにリクエスト。
IOB_tsize[1:0]	1	入力/出力	伝送サイズ=4/8/16/32/64/128/256/512/1024/2048/4096/8192/16384/32768/65536/131072/262144/524288/1048576/2097152/4194304/8388608/16777216/33554432/67108864/134217728/268435456/536870912/1073741824/2147483648/4294967296/8589934592/17179869184/34359738368/68719476736/137438953472/274877906944/549755813888/1099511627776/2199023255552/4398046511104/8796093022208/17592186044416/35184372088832/70368744177664/140737488355328/281474976710656/562949953421312/1125899906842624/2251799813685248/4503599627370496/9007199254740992/18014398509481984/36028797018963968/72057594037927936/144115188075855872/288230376151711744/576460752303423488/1152921504606846976/2305843009213693952/4611686018427387904/9223372036854775808/18446744073709551616/36893488147419103232/73786976294838206464/147573952589676412928/295147905179352825856/590295810358705651712/1180591620717411303424/2361183241434822606848/4722366482869645213696/9444732965739290427392/18889465931478580854784/37778931862957161709568/75557863725914323419136/151115727451828646838272/302231454903657293676544/604462909807314587353088/1208925819614629174706176/2417851639229258349412352/4835703278458516698824704/9671406556917033397649408/19342813113834066795298816/38685626227668133590597632/77371252455336267181195264/154742504910672534362390528/309485009821345068724781056/618970019642690137449562112/1237940039285380274899124224/2475880078570760549798248448/4951760157141521099596496896/9903520314283042199192993792/19807040628566084398385987584/39614081257132168796771975168/79228162514264337593543950336/158456325028528675187087900672/316912650057057350374175801344/633825300114114700748351602688/1267650600228229401496703205376/2535301200456458802993406410752/5070602400912917605986812821504/10141204801825835211973625643008/20282409603651670423947251286016/40564819207303340847894502572032/81129638414606681695789005144064/162259276829213363391578010288128/324518553658426726783156020576256/649037107316853453566312041152512/1298074214633706907132624082305024/2596148429267413814265248164610048/5192296858534827628530496329220096/10384593717069655257060992658440192/20769187434139310514121985316880384/41538374868278621028243970633760768/83076749736557242056487941267521536/166153499473114484112975882535043072/332306998946228968225951765070086144/664613997892457936451903530140172288/13292279957849158729038070602803456/26584559915698317458076141205606912/53169119831396634916152282411213824/106338239662793269832304564822427648/212676479325586539664609129644855296/425352958651173079329218259289710592/850705917302346158658436518579421184/1701411834604692317316873037158842368/3402823669209384634633746074317684736/6805647338418769269267492148635369472/13611294676837538538534984297270738944/27222589353675077077069968594541477888/54445178707350154154139937189082955776/108890357414700308308279874378165911552/217780714829400616616559748756331823104/435561429658801233233119497512663646208/871122859317602466466238995025327292416/1742245718635204932932477990050654848832/3484491437270409865864955980101309697664/6968982874540819731729911960202619395328/13937965749081639463459823920405238790656/27875931498163278926919647840810477581312/55751862996326557853839295681620955162624/111503725992653115707678591363241904325248/223007451985306231415357182726483808650496/446014903970612462830714365452967617300928/892029807941224925661428730905935236001952/1784059615882449851322857461811870472003904/3568119231764899702645714923623740944007808/7136238463529799405291429847247481888015616/14272476927059598810582859694494963776031232/2854495385411919762116571938898992755206256/5708990770823839524233143877797985510412512/11417981541647679048466287755595971020825024/22835963083295358096932575511191942041650048/45671926166590716193865151022383884083300096/91343852333181432387730302044767768166600192/182687704666362864775460604089535536333200384/365375409332725729550921208179071072666400768/730750818665451459101842416358142145332801536/1461501637330902918203684832716284290665603072/2923003274661805836407369665432568581331206144/5846006549323611672814739330865137162662412288/11692013098647223345629478661730274325324824576/23384026197294446691258957323460548650649649152/46768052394588893382517914646921097301299298304/93536104789177786765035829293842194602598596608/187072209578355573530071658587684389205197193216/374144419156711147060143317175368778410394386432/748288838313422294120286634350737556820788772864/1496577676626844588240573268701475113641577545728/2993155353253689176481146537402950227283155091456/5986310706507378352962293074805900454566310182912/11972621413014756705924586149611800909132620365824/23945242826029513411849172299223601818265240731648/47890485652059026823698344598447203636530481463296/95780971304118053647396689196894407273060962926592/191561942608236107294793378393788814546121925853184/383123885216472214589586756787577629092243851706368/766247770432944429179173513575155258184487703412736/1532495540865888858358347027150310516368975406825472/3064991081731777716716694054300621032737950813650944/6129982163463555433433388108601242065475901627301888/12259964326927110866866776217202484130951803254603776/24519928653854221733733552434404968261903606509207552/49039857307708443467467104868809936523807213018415104/98079714615416886934934209737619873047614426036830208/196159429230833773869868419475239746095228852073660416/392318858461667547739736838950479492190457104147320832/784637716923335095479473677900958984380914208294641664/1569275433846670190958947355801917968761828416589283328/3138550867693340381917894711603835937523656831178566656/6277101735386680763835789423207671875047313662357133312/12554203470773361527671578846415343750094627324714266624/25108406941546723055343157692830687500189254649428533248/50216813883093446110686315385661375000378909298857066496/100433627766186892221372630771322750007578185597714132992/200867255532373784442745261542645500015156371195428265984/401734511064747568885490523085291000030312742390856551968/803469022129495137770981046170582000060625484781713113936/1606938044258990275541962092341164000121250969563426227872/3213876088517980551083924184682328000242501939126845255744/6427752177035961102167848369364656000485003878252536911488/12855504354071922204335696738729312000970007756505073822976/25711008708143844408671393477458624001940015513010147645952/51422017416287688817342786954917248003880031026020295291904/102844034832575377634685573909834496007760062052040590583808/205688069665150755269371147819668992015520124104081181167616/411376139330301510538742295639337984031040248208162362335232/822752278660603021077484591278675968062080496416324724670464/1645504557321206042154969182557351936124160992832649449340928/3291009114642412084309938365114703872248321957665298898801856/6582018229284824168619876730229407744496643915330597797603712/13164036458569648337239753460458815488993287830661195595207424/26328072917139296674479506920917630977986575661322391190414848/52656145834278593348959013841835261955973151322644782380829696/105312291668557186697918027683670523911946302645289564761545312/210624583337114373395836055367341047823892605290579129523090624/421249166674228746791672110734682095647785210581158259046181248/842498333348457493583344221469364191295570421162316518092362496/1684996666696914987166688442938728382591140842324623136184724992/3369993333393829974333376885877456765182281684649246273374449984/6739986666787659948666753771754913530364563369298492546748899968/13479973333575319897333507543509827060729126738596985093497799936/26959946667150639794667015087019654121458253477193970186995599872/53919893334301279589334030174039308242916506954387940373991199744/10783978666860255917866806034807861648583301390877588074798239728/21567957333720511835733612069615723297166602781755176149596479456/43135914667441023671467224139231446594333205563510352299192958912/86271829334882047342934448278462893188666411127020704598385917824/172543658669764094685868896556925786377332822254041409196771835648/345087317339528189371737793113851572754665644508082818393543671296/690174634679056378743475586227703145509331289016165636787087342592/1380349269358112757486951172455406291018662578032331273574174685184/2760698538716225514973902344910812582037325156064662547148349370368/5521397077432451029947804689821625164074650312129325094296698740736/11042794154864902059895609379643250328149300624258650188593397481472/22085588309729804119791218759286500656298601248517300377186794962944/44171176619459608239582437518573001312597202497034600754373589925888/88342353238919216479164875037146002625194404994069201508747179851776/176684706477838432958329750074292005250388809988138403017494359703552/353369412955676865916659500148584010500777619976276806034988719407104/706738825911353731833319000297168021001553399952553612069977438814208/1413477651822707463666638000594336042003106799905110724139954877628416/2826955303645414927333276001188672084006213999102221482799097555256832/5653910607290829854666552002377344168012427998204442965598195110513664/11307821214581659709333104004754688336024855996408885931196390221027328/22615642429163319418666208009509376672049711992817771862392780442054656/45231284858326638837332416019018753344099423985635543724785560884109312/90462569716653277674664832038037506688198847971271087449571121768218624/180925139433306555349329664076075013376397695942542174899142243536437248/361850278866613110698659328152150026752795391885084349798284487072874496/723700557733226221397318656304300053505590783770168699596568974145748992/1447401115466452442794637312608600107011181567540337399193137948291497984/2894802230932904885589274625217200214022363135080674798386275896582995968/5789604461865809771178549250434400428044726270161349596772551793165991936/1157920892373161954235709850086880085608945254032269919354510358633198384/2315841784746323908471419700173760171217890508064539838709020717266396768/4631683569492647816942839400347520342435781016129079677418041434532793536/9263367138985295633885678800695040684871562032258159354836082869065587072/18526734277970591267771357601390081369743124064516318709672165738131174144/37053468555941182535542715202780162739486248129032637419344331476262348288/74106937111882365071085430405560325478972496258065274838688662952524696576/148213874223764730142170860811120650957944992516130549677377325905049393152/296427748447529460284341721622241301915889985032261099354754651810098786304/592855496895058920568683443244482603831779970064522198709509303620197572608/1185710993790117841137366886488965207663559940129044397419018607240395145216/2371421987580235682274733772977930415327119880258088794838037214480790290432/4742843975160471364549467545955860830654239760516177589676074428961580580864/9485687950320942729098935091911721661308479521032355179352148857923161161728/18971375900641885458197870183823443322616959042064710358704297715846322323456/37942751801283770916395740367646886645233918084129420717408595431692644646912/75885503602567541832791480735293773290467836168258841434817190863385289293824/151771007205135083665582961470587545780935672336517682869634381726770578587648/303542014410270167331165922941175091561871344673035365739268763453541157175296/607084028820540334662331845882350183123742689346070731478537526907082314350592/1214168057641080669324663691764700366247485378692141462957075053814164628701184/2428336115282161338649327383529400732494970757384282925914150107628329257402368/4856672230564322677298654767058801464989941514768565851828300215256658514804736/9713344461128645354597309534117602929979883029537131703656600430513317029609472/19426688922257290709194619068235205859959766059074263407313200861026634059218944/3885337784

クは応用（メモリ／PCI／キャッシュ等）によって、ユニット毎に異なる。しかし、実際のバス仲裁ユニットは各ユニットに対し同一であり、すべてのサブモジュールで重複される。このユニットは、外部バスマスタ／スレーブと内部ユニットロジック間の媒体として作用する。例えば、メモリコントローラーの場合、一応CASが活性化されると、メモリコントローラーは、FBUSを使用する必要があることを表わす内部信号を通して、内部リクエストをFBUS仲裁ロジックに表示する。このリクエストに応信し、FBUSコントローラーは、メモリコントローラーに対して外部のシステムにリクエストを表示し、承認を待つ。一応承認が受信されると、アドレス／データ制御は、応信の第1番目のエントリーとメモリコントローラーのデータFIFOから伝送される。

【0084】メモリコントローラーに対するシステムリクエストサイズは、1バイトから最大32ビットサイズまでできる。32バイト以上のリクエストサイズの場合 *

リクエストID [9:6]

0000

0001

0010

0011

0100

0101

0110

0111

1xxx

目的地IDビット [2] は、読出／記入リクエストステータスを表わすことに使用される。これはFBUSがアドレスリクエスト（読出）と、アドレス／データリクエスト（記入）間を区別することを助けてやる。正常状態で、承認カウンタビット“grCNT [1:0]”は、リクエストがバスを必要とするFBUSサイクル数を示す。折り返しリクエストに対し、リクエストはバスマスタにリクエストの長さを知らせる。FBUSマスタコントローラーは2個の承認カウンタビットによって承認を表示する。FBUSはポストされた読出を支援するスプリットランザクションバスである。これはリクエストがバスをリクエストし、一応承認されると、このFBUSはアドレスを駆動しランザクションを終了する。しばらく後で、スレーブ／データソースは目的地IDを※

* 合、ソース／リクエストはFBUSサイズビットを使用し、多数個のリクエストを初期化する。これはSDRAMメモリバス（1または2個の（三星 SDRAM 1M*16）の限界に因ることである。SDRAMは残りのシステムにより要求される完全な32バイトを実現するために、8個のラップ(wrap)の長さに対してプログラムされる。32バイト以下のリクエストの場合、32バイトのすべてがSDRAMからフェッチされるが、所望の数のバイトのみが目的地に伝送される。また、10個のビットリクエストIDバスは、“チップ選択”信号（アドレス／データと同一なサイクル）で有効化される。すべてのFBUSノードは、3ビットの“目的地ID”をFBUSアービタに発生する。この3ビットはリクエストと共に有効化され、リクエストの目的地を表わす。目的地IDビット [1:0] は、下記のように入力されるリクエストIDからデコーディングされる。

【0085】

ソース	目的地ID [1:0]
予約	N/A
ARM7	N/A
FU	N/A
LSU	N/A
CCU	00
ASIC	11
MEM	01
PCI	10
予約	

※使用し、かつ同一なリクエスト112をリクエストに戻すことによってデータを戻す。このような特性は、バス帯域幅を大幅向上させ、他のマスタがFBUSのさらに迅速な使用を許容する。より詳細なことはタイミング図を参照すること。

4. 2 ビン説明

以下、システムFBUSのアドレス、データ及び制御信号を説明する。上述のごとく、FBUSは非多重化されたアドレス／データバスである。“xxx”はリクエスト名称(mem、pci、asc、ccu)を表わす3個の文字コードである。

【0086】システムFBUS信号定義

【表29】

システムFBUS信号定義

信号名	サイズ	方向	コメント
Faddr[31:0]	32	両方向	別途のアドレスライン上の32ビットアドレスバス
Fdata[63:0]	64	両方向	ダブルポンピングされた64ビットデータバス
Frd_wr_1	1	両方向	“1”==>読出、“0”==>記入
Fxxx_cs_1	1	入力	チップセレクト。アクチブロー
Frdy_i	1	両方向	データ準備信号、アクチュアルデータの前に有効な一サイクル。データの確認によりのみ可能である。アクチブロー。
Freq_ID	10	両方向	データに先立つリクエストID。このような信号は、Fdrdy信号ビット[9:6]--->リクエストIDビット[5:0]--->トランザクションIDと同一なタイミングを有する。
Fxxx_req_1	1	出力	ソースFbusリクエスト。アクチブロー。
Fxxx_grCNT	2	出力	サイクルグラントが必要でないことを表わすためにリクエストで有効にするFbusグラントをカウントする。
Fxxx-grant-1	1	入力	中心Arb。ユニットからFbusグラント。アクチブロー。
Fxxx_did	3	出力	Fbus目的地ID。この信号はCSが発生されたことをメインFbusアービタに表わすようにするリクエストのある場合に駆動される。 Fxxx_did[1:0]==目的地ID Fxxx_did[2]==読出/記入状態 (0=>記入、1=>読出)

【表30】

Fxxx_afull	1	出力	アドレスfifoが満たされる。この信号は、fifoに(n-1)有効エントリが存在する時確認される。アクチブハイ
Fxxx_dfull	1	出力	データfifoが満たされる。この信号は、fifoに(n-1)有効エントリが存在する時確認される。アクチブハイ
f_reset_1	1	入力	アクチブロー同期Fbusリセット
Fclk	1	入力	80MHzバスクロック
Fpr_wr_1	1	入力	部分記入表示器。受信器はこの信号を利用し、それぞれの8バイト内にあるバイトのサイズを決定できるようになる。この信号はFreq size[7:0]と共に使用される。
Freq_size	8	両方向	伝送サイズ(<16バイト、32バイト、64バイト、128バイト) *。*RMWのためにデコードテーブル定義が必要である。

【0087】図30は、メモリ読出リクエストFBUS スレーブ“非メモリ”リクエストFBUSフローを示しているし、図32はマスタ/スレーブ“非メモリ”リクエストFBUSフローを示したものであり、図31はメモリ記入リクエ 50

たものであり、図33は中央のFBUS仲裁制御ユニットを示したものである。図34～図36はFBUSタイミング図であり、図34はメモリリクエストFBUSタイミングを示す（8バイトデータ伝送を示しており、16/32/64/128バイトの複数のデータサイクルが使用される）。図35はメモリ読出リクエストFBUSタイミングを示し（伝送サイズ=8バイト）、図36はメモリ折り返し記入リクエスト（伝送サイズ=32バイト）を示したものである。

【0088】第5章 PCIバス

本章は、PCIコアと、内部FBUSとインターフェースするPCIグルー(glue)ロジックを説明したものである。

5.1 概要

MSP_1E PCIコントローラーは、PCIバスベック改訂版2.1を満足させるために設計されたものである。より詳細なことはこの標準スペックを参照すること。PCIユニットは、2個のメインセクション：PCIコアとFBUS‘グルー’ロジックを含む。PCIコアは、主に33MHzのPCIバス速度で動作する外部のPCIデバイスとインターフェースする。FBUS‘グルー’ロジックは、80MHzで動作する三星FBUSとインターフェースする。この‘グルー’ロジックは、PCIコアとFBUS間をインターフェースする。速度同期化は、サブブロックの2個のエンドでFIFOを利用して実現できる。三星のPCIコアは、また仮想的なフレームバッファロジックと、FBUSを通してARM7とインターフェースすることに必要なすべてのVFBレジスタを含む。このPCIユニットに対し唯一な特徴は、ホストCPU MSPチップと、MSPチップからホストCPUへのインタラプト処理である。これに対してより詳細に説明する。

【0089】5.1.1 三星PCIコアブロック図は図37の図示のとおりである。

5.2 PCI FBUSインターフェースロジック（図38参照）

PCIコアのサブブロックは、MSP内部FBUSとSANDマイクロのPCIコアとインターフェースする。アドレスとデータは、2個のエンドでFIFOに貯蔵される。このサブブロックはまた、PCI信号とFBUSクロックを同期化させる役割をする。PCIコアロジックは、FBUSマスタ及びスレーブデバイスであることもある。大部のアクセスは、64ビットFBUSを通してローカルSDRAMメモリに向かう。FBUSプロトコルの説明のためには、FBUS章を参照すること。PCI FBUS制御ロジックは、また仮想フレームバッファレジスタと制御とを含む。このレジスタはFBUSを通してARM7によりプログラムされる。ブロック図を参照すること。

5.3 PCI VFBロジック

図39はVFBブロック図であり、図40はVFBレジスタである。

5.4 PCIコアロジック

MSP PCIコアは、PCI2.1スペックを完全に満足する。追加事項はインタラプトとソフトウェアMSPリセットの為に付加されたレジスタ数である。ARM7にあるソフトウェア、MSP制御レジスタのMSP (bit<3>) からPCIホストインタラプトリクエストをセットすることによって、ホストCPUをインタラプトすることができる。これはPCIバス(INTA#)にあるインタラプトピンをセットすることによって、PCIコアロジックがホストCPUをインタラプトするようにする。以降、ホストCPUは、MSP制御レジスタのPCIホストインタラプト認知 (bit<4>) を通してインタラプトを認知する。これはインタラプトラインが非活性状態となるようにする。MSP PCIコアはまた基本的に、ARM7に対するインタラプトであるホストCPUからのインタラプトを受け取ることができる。PCIスペックが任意のインタラプト入力ピンを支援しないので、MSP制御レジスタにある、ホストからのMSPインタラプトリクエスト (bit<2>) がこの機能を提供することに使用される。ホストCPUは、ARM7に対するインタラプトを表わすためにこのビットを設定することができる。次に、一応ホストインタラプトを認知すると、ARM7はこのレジスタをクリアさせる。図41のブロック図を参照すること。図41に対して、PCI空間でないMSP領域にマッピングされた3個のレジスタが必要である。実質的なPCIコアに対する細部的な情報のためには、PCI2.1スペックを参照すること。

【0090】第6章 メモリコントローラー

6.1 本章は、ハードウェアとソフトウェアデザイナー側面で、メモリコントローラーの仕様を説明したものである。

6.2 概要

MSPメモリコントローラーはいく特徴を有し、費用と性能に対するトレードオフのためのプログラム可能性レベルを有する。メモリコントローラーは80MHzで動作するメインシステムバス“FBUS”とDRAMチップとインターフェースする。80MHzクロック周波数を実現するために、初期の設計段階で同期DRAMが使用される。結局、メモリサブシステムは、標準高速ページDRAM、拡張されたデータ出力(EDO)DRAMと同期DRAMとを支援する。メモリバンクサイズは、インターリーブ可能な2個の外部バンクに制限される。初期の同期DRAMメモリコントローラーは、DRAMを動作させることに必要な最小限の特徴を有する。次は基本的な第1パスメモリコントローラーの特徴を示している。

50 *三星の同期DRAM支援

- * 2個のSDRAMチップを使用した1つのメモリバンク (1M×16)
- * Cas-Before-Ras (CBR) リフレッシュ支援
- * 読出-修正-記入 (Read-Modify-Write) 動作を初期化する部分的な記入支援
- * 内部のバンクインターリーブ支援 (MA [11] を通したピンポン)
- * 80MHzメモリとプロセッサバス (1:1) 周波数マッチ
- * プログラマブルリフレッシュ率
- * システムバスを効率的に使用するためのアドレスとデータキューイング
- * マニュアル“2個バンクプリチャージ”支援

MSPメモリコントローラーは、2個のメインサブ構成要素: データコントローラーとアドレスコントローラーとを有する。データコントローラーは、DRAMから読み出されたデータを貯蔵し、プロセッサバスからデータを記入するための読出及び記入データキューを有する。データコントローラーはまた、バイト記入のためのRMWロジックを含む。データコントローラーに対するすべての制御は、アドレスコントローラーから発生する。アドレスコントローラーは、リクエストキュー、応信IDキュー、メモリアクセスデコーディングロジック、ページ比較器ロジック、RAS/CAS状態マシン、リフレッシュ状態マシンと、データコントローラーにより使用される必要なすべての制御信号を有する。SDRAMメモリクロックは、システムクロックと同一である。SDRAMは前記1セットの各制御信号を受信する。

【0091】6. 2. 1 メモリコントローラーブロック図は図42の図示のとおりである。

6. 2. 2 メモリコントローラーフローは、図43の図示のとおりである。

6. 3 アドレスコントローラー (AC)

メモリコントローラーでアドレスコントローラーセクションは、データコントローラーを管理することだけでなく、すべてのDRAM制御を発生させる役割をする。MSPメモリコントローラーのこのセクションは、またFBUSインターフェースのアドレスと制御経路を担当する。次のブロック図は、アドレスコントローラーユニットの多数個のサブセクションを示す。

6. 3. 1 アドレスコントローラーブロック図は図44の図示のとおりである。

6. 3. 2 メモリコントローラーリクエストFIFO

MSPメモリコントローラーは、実質的なメモリコント*

- * ローラー状態マシンへのディスパッチ(dispatch)のための、FBUSアドレスと制御情報とを貯蔵する4個のディープリクエストFIFOを有する。リクエストFIFOのそれぞれのエンタリーは、特定エンタリーが有効であることを表わす“有効”ビットを有する。メモリコントローラー状態マシンは、常にENTRY_0のFIFOにある最下位エンタリーを支援する。一応リクエストが提供され、列アドレスストロブ(CAS)が活性化されると、メモリコントローラーはこのエンタリーをクリアーさせるために、クリアー信号を表示する。FIFO FULL/EMPTYステータスによって、パレルシフトが有効な内容をエンタリー0にシフトさせるために初期化される。MSPメモリコントローラーリクエストFIFOフォーマットは、図45の図示のとおりである。

【0092】6. 3. 3 メモリコントローラーアドレスデコード/マップ

アドレスデコーディングロジックは、主に11ビットのSDRAM行アドレスMA [10:0]と8ビットの列アドレスMA [7:0]とを発生させる役割をする。このアドレスラインは、SDRAMアドレス入力 [11:0]へ直接駆動される。メモリアドレスビット [11]は、性能のために内部SDRAMバンクと、改善されたメモリバス使用の間をトグルすることに使用される。このメモリアドレスは、次のことを表わすレジスタを通して与えられるプログラマブルマルチプレクサーを使用して発生される。

—現在システムキャッシュラインサイズ

—内部バンクの数

—内部バンクインターリーブピン

システムキャッシュラインオフセットは、32バイトキャッシュラインに対して5ビットである。図46は、16MB DRAMのためのFBUSシステムアドレスから発生される提案されたメモリアドレスフォーマットを示している。この多重化されたメモリアドレスは、メモリコントローラー状態マシンによって指示されるRASとCASストロブとを有する、1つのサイクルに対して有効である。MCUは読出-修正-記入動作を指示せず、8バイト記入を遂行することができる。しかし、FBUSアドレスのbit [2]は常にアドレスのみをスタートするために、ゼロである。このビットは下記のように入スタートアドレスを表わす3個ビットの中の1つである、SDRAMアドレスのbit [0]にマップされる。

Faddr [4:2]

000

010

100

110

記入シーケンス (WRAP=8)

0-1-2-3-4-5-6-7

2-3-4-5-6-7-0-1

4-5-6-7-0-1-2-3

6-7-0-1-2-3-4-5

これらは全部偶数のスターティングアドレスであり、MCUによって支援されるシーケンスである。すべての読出動作は32バイトを仮定し、スタートアドレスは(000)= $rna[2:0]=Faddr[4:2]$ である。

【0093】6.3.4 メモリコントローラ状態マシン

MSPメモリコントローラは、1つのマスタコントローラ状態マシンを有する。この状態マシンは、SDRAM制御信号のためのすべてのタイミング(RAS/CAS/WE/CS/DQM)を発生させる役割を担う。状態マシンは、常にエントリー0にある有効エントリーのために、リクエストFIFOをモニタする。一応、有効ビットが検出されると、状態マシンはSDRAMシーケンス開始をキックオフする。また、RASプリチャージが必要であるか否かを判断するために、ページ比較器からPage_hit信号をモニタする。RASプロチャージは、現在のアクティブ/開放バンク上で遂行される。マニュアルプリチャージシーケンスは、ゼロの状態*

SDRAMパラメータ

DRAM	tRAS	tRRD	tCCD	tRCD	tCAC	tRP
80MHz (12.5ns) -10Part	6Cycles =75ns	2cycles	1cycle	2cycles	3cycles CAS Laten.	3cycles
83MHz (12ns) -12Part	6cycles	2cycles	1cycle	2cycles	3cycles CAS Laten.	3cycles

【0095】* tRASは、同期DRAMの60ns列アクセスタイムを実現させるために、5サイクルで使用され得る。メモリコントローラタイミング図を参照すること。

6.3.4.1 状態マシンダイアグラム

図47はSDRAMメモリコントローラRAS/CAS状態マシンダイアグラムを示す。

6.4 メモリコントローラリフレッシュ

同期DRAMは、それぞれの貯蔵セルにあるデータを維持するために毎32

ms(15.6 μ s)毎にリフレッシュされる必要がある。同期DRAMはまた、2個モードのリフレッシュ：自動リフレッシュとセルフリフレッシュとを支援する。

【0096】6.4.1 SDRAM自動リフレッシュ
標準の自動リフレッシュを使用し、2個の内部バンクが内部カウンタにより交替的にリフレッシュされる。行(row)の数が4096であるので、自動リフレッシュはDRAM全体をリフレッシュするために、2048自動リフレッシュサイクルを必要とする。自動リフレッシュコ

*を活性化させるために、CS、RAS、WEとMA[11]を表示することを含む。内部バンク選択ビットMA[11]は、プリチャージするためのバンクを選択することに使用される。読出の場合：プリチャージコマンドは、データ衝突を避けるために、データがSDRAMから受信されてから表示される。記入の場合、プリチャージは最後のビットのデータがメモリに記入されてから発行される。一応プリチャージコマンドが完了されると、特定バンクは次のメモリ動作のためにアイドル(idle)状態となる。SDRAMスペックによると、プリチャージコマンドは、tRAS(min)(ここでは60ns)が満足されてから何時でも発生できる。しかし、現在4のラップ(wrap)の長さによって、メモリコントローラ状態マシンは、データがメモリに読出/記入されてからプリチャージコマンドを発生させる。次は、MSPメモリコントローラと共に使用されるSDRAMパラメータを示す。

【0094】

【表31】

マンドは、CKEとWEがハイで、CS、RAS&CASがローであることを表示することにより発生される。このコマンドは、2個のバンクがアイドル状態にある場合のみに表示される。自動リフレッシュを終了することに必要な時間は、

$t_{RC}(\min)/\text{サイクル時間}=100\text{ns}(\text{spec})/12.5\text{ns}=8\text{サイクル}(80\text{MHz})$

6.4.2 SDRAMセルフリフレッシュ

セルフリフレッシュは、三星のSDRAMに使用されるさらに別のモードである。これは一般的にデータ維持及び低電力動作のために好ましいリフレッシュモードである。ここでSDRAMは、内部クロックとCKEを除いたすべての入力バッファをディスエーブルさせる。CS、RAS、CASとCKEがローでWEがハイの場合、セルフリフレッシュモードに入る。セルフリフレッシュモードは、SDRAMクロックのシュテティングとCKE信号を用いた再試みを要求するので、MSPメモリコントローラは、このリフレッシュモードを使用しない。

6. 4. 3 マニュアルリフレッシュ

このリフレッシュモードは、状態マシン/カウンタ設計を要求する。カウンタは15.6 μ s毎にタイムアウトされ、メモリコントローラロジックでリフレッシュストロブを表示する。次に、メモリコントローラは、現在のリフレッシュを終了し、すぐSDRAMリフレッシュサイクルを初期化させる。このサイクルは、アイドル状態における制限を持たず、全く自動リフレッシュサイクルと類似している。

【0097】6. 5 データコントローラ (DC)

メモリコントローラでデータコントローラセクションは、主にプロセッサからデータを記入するか、またはSDRAMからデータを読み出すためのデータキューとして提供される。このコントローラはまた、すべての部分的な記入時 (バイト記入) のための記入併合ロジックを有する。部分的な記入はまず、DRAM読出をキックオフした後データを併合し、最後に完全に修正されたワードをメモリに更に記入する。従って、部分的な記入シーケンス次の任意のリクエストは性能ヒットを取らなければならない。

6. 5. 1 データコントローラブロック図は図48の図示のとおりである。

【0098】6. 6 ピン説明

このコントローラは、次のパッケージピンを提供する。

*RAS_I : 出力ピン (アクチブロー)。これはMA [11:0] からの行アドレスを、選択されたDRAMバンクの内部行アドレスバッファにラッチするための行アドレスストロブである。

*CAS_I : 出力ピン (アクチブロー)。これはMA [11:0] からの列アドレスを、選択されたDRAMバンクの内部列アドレスバッファにラッチするための、列アドレスストロブである。

*WE_I : 出力ピン (記入時アクチブロー)。DRAMの記入イネーブル入力ピンを駆動するためのものである。

*MA [11:0] : 出力ピン。DRAMに対する多重化された行及び列アドレス信号。

*DQM : 出力ピン。クロック及び出力をマスクした後、SDRAMデータ出力をハイインピーダンスにする。(このピンは同期DRAMインターフェースに対してのみ使用する。)

*CS_I : 出力ピン (アクチブロー)。選択されたSDRAM動作のためにディスエーブルまたはイネーブルされる。(このピンは同期DRAMインターフェースに対してのみ使用する。)

*CLK : 出力ピン。これは同期DRAMに対するクロック出力ピンであって、SDRAMのみで使用され、MSPのシステムクロックと同じ位相を有する。

【0099】6. 7 メモリコントローラタイミング

図は、図49から図51の図示のとおりである。図49に関連した事項は下記のとおりである。

—三星のSDRAMに仮定

—80MHzで動作するメモリとシステム。

—1個または2個の外部SDRAM (1M \times 16)。

—メモリからラインをフェッチするための4/8プログラマブルラップの長さ。

—tRCD=3。

—tCAS=3。

—内部遅延=2クロック。

—メモリ待ち時間=8サイクル (8 \times 12.5=100ns)。

—メモリからのシステムデータは、仲裁 (読出データ) のために2個サイクルほど遅延する。

【0100】6. 8 プログラマブルモデル

プログラマー側面で、メモリコントローラに関連した制御レジスタは、下記のとおりである。

6. 8. 1 SDRAMリセットレジスタ (R/W)

このレジスタは、それぞれのシステムリセット後でリセットされる。これはSDRAMパワーオンシーケンスを始めるreset_sdram信号を伝達する1ビットレジスタである。システムリセット時にこのレジスタは1に設定される。SDRAMを動作させるために、ソフトウェアによりこのレジスタをクリアさせなければならない。bit0はシステムリセットで設定され、SDRAMを動作させるためにクリアされる。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b1011

6. 8. 2 SDRAMバーストタイプレジスタ (R/W)

このレジスタは、SDRAMバーストタイプをプログラムする。これは順次的なバーストタイプに対しゼロにプログラムされる1ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b1010

bit0はシステムリセットとともにリセットされ、SDRAMを動作させるためにクリアされる。

6. 8. 3 SDRAMリフレッシュレジスタ (R/W)

このレジスタは、SDRAMリフレッシュ値をプログラムする。これはFBUSを通してプログラムされる12ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b1001

bit11-0はシステムリセットとともにリセットされ、4E0のリフレッシュ値にプログラムされる。

6. 8. 4 SDRAM RASプリチャージ (tRP)

レジスタ (R/W)

このレジスタはSDRAM RASプリチャージ値をプログラムする。これはFBUSを通してプログラムされる3ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b1000

bit 2-0はシステムリセットとともにリセットされ、1または2または3にプログラムされる。

6. 8. 5 SDRAM CAS待ち時間 (tCAC) レジスタ (R/W)

このレジスタはSDRAM CAS待ち時間をプログラムする。これはFBUSを通してプログラムされる3ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b0011

bit 2-0は、システムリセットとともにリセットされ、1または2または3にプログラムされる。

6. 8. 6 SDRAM RAS CAS待ち時間 (tRCD) レジスタ (R/W)

このレジスタはSDRAM RCD待ち時間をプログラムする。これはFBUSを通してプログラムされる3ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b0010

bit 2-0は、システムリセットとともにリセットされ、1または2または3にプログラムされる。

6. 8. 7 SDRAM WRAP LENGTHレジスタ (R/W)

このレジスタはデータに対するSDRAMのラップ長さをプログラムする。これはFBUSを通してプログラムされる3ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b0001

bit 2-0は、システムリセットとともにリセットされ、1、2、4、または8の中にいずれかにプログラムされる。

6. 8. 8 SDRAM NOP TIMEレジスタ (R/W)

このレジスタはパワーオンシーケンスのためのSDRAMのNOP時間をプログラムする。これはFBUSを通してプログラムされる16ビットレジスタである。

プログラミングアドレス:

Faddr [31:20] = 12' h010

Faddr [3:0] = 4' b0000

bit 15-0はシステムリセットとともにリセットされ、クロック周波数によって200usにプログラムさ

れる。

【0101】第7章 ASICインターフェース
本章ではASICインターフェースユニットの仕様を説明した。

7. 1 概要

ASICインターフェースユニット (図52参照) は、1つのプログラマブル32ビットDMA、多数個のFIFOと制御ブロックを有する。ASICインターフェースブロックは、80MHzで動作するメインシステムバス (FBUS) と、MSP、AD1843 (オーディオ、電話)、KS0122 (ビデオキャプチャ)、KS0119とVGAをインターフェースするCODECインターフェースブロックをインターフェースする。現在の仮定は、任意の同期化問題を避けるために、すべてのCODECインターフェースとDMAコントローラーを完全なFBUS速度で動作させることである。カスタムASICブロックは、3個の主要セクション: FBUSマスタ/スレーブインターフェース、MSP 8-チャンネルDMAコントローラーと実際のCODECとを有する。データはFBUSからCODECに伝達されるか、またはCODECからFBUSに伝達される。しかし、アドレスはただDMAコントローラーのみから発生される。そうすると、このアドレスはFBUSインターフェースロジックでマッピングされたFBUSである可能性もある。他のFBUSノードからのすべての記入は、ただCODECセクションにあるレジスタのみをプログラムする。他のすべてのトラヒックでは、サイズ及びID情報を有する応信を読み出さなければならない。FBUS仕様を参照すること。次は、ASICインターフェースユニットに対する特徴である。

* 32ビット基本DMA機能を支援 (各コーデックに対し、8個のチャンネル-1個のチャンネル)。

* 2個の4ディープ×64ビットデータFIFO。

* 1個の1ディープ×52ビットリクエストFIFO。

* 1個の2ディープ×52ビット応信FIFO。

* FBUSとCODECインターフェースブロックのためのマスタ/スレーブを支援。

* 動作周波数: 80MHzまで。

* メモリに対するIO、IOに対するメモリ間のアクセス支援。

* KS0119用に使用されるチャンネル0のための最上位優先順位支援。

* KS0119に対して高性能を実現するために特殊アドレスバス支援。

このカスタムインターフェースロジックは、3個の相互に異なるCODECを支援する。

* オーディオ及び電話CODEC (AD1843)。このCODECはDMAコントローラーと通信する両方向64ビットデータバスを有する。(チャンネル4→DAC1、チャンネル5→DAC2、チャンネル6→ADC

左側、チャンネル7→ADC右側)

*ビデオキャプチャCODEC (KS0122)。このCODECは両方向64ビットデータバスを有し、DMA (チャンネル2) に対してM→IO、IO→Mリクエストを初期化することができる。

*ビデオ バックエンド(backend) CODEC (KS0119)。このCODECはメモリコントローラーからデータを直接に受信する (チャンネル0)。

【0102】ASICインターフェースブロック

7. 2 直接メモリアクセス (DMA) コントローラー 10
DMAコントローラーは、アドレス発生及び解釈のために使用されるレジスタを有する。このDMAコントローラーは、8個の独立的なチャンネルを有する。各チャンネルは、現在のアドレスレジスタと停止アドレスレジスタとを有する。開始及び停止アドレスレジスタは、構成ブロックを通して先にプログラムされる。現在のアドレスレジスタは、8個のCODECの中のいずれか1つからDMAリクエストが発生する時ごとにロードされる。一応、FBUSがアクセスを承認すると、このDMAアドレスは、現在アドレスが停止アドレスレジスタとマッ 20
チされるまで、サイクルごとに増加する。その時点で、DMAコントローラーが信号“EOP (End Of Process)”を発生する。この信号はプロセスでインタラプトを誘発する。8個のすべてのDMAチャンネルは、マルチプレクサーとアドレス比較ブロックを制御する共通の仲裁ユニットを有する。このDMAコントローラーは、I/Oメモリ、メモリとIO、メモリとメモリ間のアクセスを支援する。CODECがDMAと通信しようとする時ごとに、CODECはDMA_REQ信号を表示し、DMAからDMA認知信号の“DACK”を待つ。一応、 30
認知されるとCODECはM→IO信号とデータとを駆動する。DMAコントローラーは承認されたDACKによって、適切なチャンネルを選択する。ブロック図を参照すること。

【0103】7. 3 DMAレジスタ説明

7. 3. 1 現在アドレスレジスタ

各チャンネルは、すべてのアドレスが8バイトに配列されることを要求する、29ビット現在アドレスレジスタ (bits<31:3>) を有する。事実上、このレジスタは29ビットカウンタである。このレジスタはARM 7によって読み出され、初期値はFBUSを通してARM 7からロードされる。このアドレスはデータ伝送サイズに基づいて増加される。現在アドレスレジスタにあるアドレスは、マルチプレクサーを通してFBUS上のアドレスをロードするために、アドレス発生ブロックに伝送される。現在アドレスレジスタは、アイドル状態ではアドレス値をホールドさせる。

7. 3. 2 停止アドレスレジスタ

各チャンネルは、すべてのアドレスは8バイトに配列されることを要求する、29ビット停止アドレスレジスタ 50

(bits<31:3>) を有する。このレジスタには、FBUSを通してARM 7により記入される。この値は、比較ブロックで現在アドレスと比較されることに使用される。もし現在アドレスが停止アドレスと一致すると、DMAコントローラーは、各チャンネルに対して“EOP”信号を発生させる。

7. 3. 3 ステータスレジスタ

このレジスタは、各チャンネルが停止アドレスに到達し 10
か否かを表わす情報を貯蔵する。Bits<7:0>は、どのチャンネルが停止アドレスに到達したかを規定し、ARM 7がCCUを通して現在アドレスレジスタを初期化した時にリセットされる。このレジスタはARM 7により読み出され、ARM 7はこのレジスタを記入することができない。

7. 3. 4 制御レジスタ

このレジスタは、DMAコントローラーの動作に対する情報を貯蔵する。Bits<7:0>は、どのDMAチャンネルが動作のためにイネーブルされたかを規定する。このビットは当該チャンネルが停止アドレスに到達する時ごとにリセットされ、ARM 7は動作を再開するためにこのビットを設定する。任意のチャンネルイネーブルビットが“0”の場合、DMAは例えばCODECがDMAにDMA_REQを送るとしても、当該CODECにDMA_ACKを送らない。Bits<19:16>は、DMAチャンネルの中のいずれか1対が、ダブルバッファとして動作するために共に連結されているかを規定する。例えば、チャンネル0とチャンネル1がダブルバッファとして連結されている場合、チャンネル0の現在アドレスが停止アドレスに到達した場合、DMAコントローラーは自動的にチャンネル1を切り換 20
え、チャンネル1の現在アドレスが停止アドレスに到達した場合、DMAコントローラーは自動的にチャンネル0を切り換える。Bit<28:21>は、各チャンネルの読出/記入モードに関連した情報を貯蔵する。もしこのビットの中の任意のビットがARM 7によって“1”に設定されると、該当チャンネルは読出動作のために使用され、残りのチャンネルは記入動作のために使用される。Bit<31>は、DMAがEOP信号をインタラプトコントローラーに送ったか否かを規定する。もしこのビットが“0”であれば、DMAは任意のチャンネルが停止アドレスに到達しても、EOPを送らない。

【0104】7. 3. 5 マスクレジスタ

制御レジスタにある各ビットは、マスクレジスタにあるマスクビットと連結されている。マスクビットが“0”であれば、制御レジスタにある該当ビットがアップデートされることを防止する。初期には、このレジスタ<31:0>は、FFFFFFFF (hex)に設定される。

7. 3. 6 プログラミング

開始及び停止アドレスは、FBUSを通してARM 7に

よりプログラムされる。FBUSマッピング値は下記のとおりである。

CCU→0040_0000-007F_FFFF、

MCU→0080_0000-047F_FFFF、

PCI→0800_0000-FFFF_FFFF。 *

DMAレジスタアドレスマップ

アドレスオフセット <26:0>(HEX)	ビット#	説 明
4A0_0000	29	現在アドレスレジスタ0
4A0_0008	29	現在アドレスレジスタ1
4A0_0010	29	現在アドレスレジスタ2
4A0_0018	29	現在アドレスレジスタ3
4A0_0020	29	現在アドレスレジスタ4
4A0_0028	29	現在アドレスレジスタ5
4A0_0030	29	現在アドレスレジスタ6
4A0_0038	29	現在アドレスレジスタ7
4A0_0040		予約
4A0_0048		予約
4A0_0050	29	停止アドレスレジスタ0
4A0_0058	29	停止アドレスレジスタ1
4A0_0060	29	停止アドレスレジスタ2
4A0_0068	29	停止アドレスレジスタ3
4A0_0070	29	停止アドレスレジスタ4
4A0_0078	29	停止アドレスレジスタ5
4A0_0080	29	停止アドレスレジスタ6
4A0_0088	29	停止アドレスレジスタ7
4A0_0090		予約
4A0_0098		予約
4A0_00A0	32	状態レジスタ
4A0_00A8	32	制御レジスタ
4A0_00B0	32	マスクレジスタ

* アドレスプログラミングにおいて、Address [26:0] は表に基づいて設定される。

【0105】DMAレジスタアドレスマップ
【表32】

【0106】

【表33】

ステータスレジスタのエンコーディング

ビット#	説 明
0	チャンネル0は停止アドレスに到達した。
1	チャンネル1は停止アドレスに到達した。
2	チャンネル2は停止アドレスに到達した。
3	チャンネル3は停止アドレスに到達した。
4	チャンネル4は停止アドレスに到達した。
5	チャンネル5は停止アドレスに到達した。
6	チャンネル6は停止アドレスに到達した。
7	チャンネル7は停止アドレスに到達した。
8	予約
9	予約
10-31	予約

【0107】制御レジスタのエンコーディング * * 【表34】

制御レジスタのエンコーディング

ビット#	説 明
0	DMA伝送のためのイネーブルチャンネル0
1	DMA伝送のためのイネーブルチャンネル1
2	DMA伝送のためのイネーブルチャンネル2
3	DMA伝送のためのイネーブルチャンネル3
4	DMA伝送のためのイネーブルチャンネル4
5	DMA伝送のためのイネーブルチャンネル5
6	DMA伝送のためのイネーブルチャンネル6
7	DMA伝送のためのイネーブルチャンネル7
8	予約
9	予約
15:10	予約
16	チャンネル0及びチャンネル1のためのダブルバッファ選択
17	チャンネル2及びチャンネル3のためのダブルバッファ選択
18	チャンネル4及びチャンネル5のためのダブルバッファ選択
19	チャンネル6及びチャンネル7のためのダブルバッファ選択
20	予約
21	読出のためにチャンネル0が使用される(ビットが'1'の場合)
22	読出のためにチャンネル1が使用される(ビットが'1'の場合)
23	読出のためにチャンネル2が使用される(ビットが'1'の場合)
24	読出のためにチャンネル3が使用される(ビットが'1'の場合)
25	読出のためにチャンネル4が使用される(ビットが'1'の場合)
26	読出のためにチャンネル5が使用される(ビットが'1'の場合)
27	読出のためにチャンネル6が使用される(ビットが'1'の場合)
28	読出のためにチャンネル7が使用される(ビットが'1'の場合)
29	予約
30	予約
31	DMAインタラプトイネーブル

【0108】7.4 CODEC初期化

カスタマASICユニットは、各CODECの初期化を

支援する。実質的には、ARM7がカスタマASICユニットを通してCODEC初期化を担当する。カスタマASICユニットは、各CODECに対するリクエスト信号を発生させるためのアドレスデコーダーを有している。カスタマASICユニットは、任意のCODECと通信しようとする時ごとに、CODECにリクエスト信号を送り、CODECからの認知信号を待つ。認知信号を受信してから、カスタマASICユニットは、データとアドレスとをCODECに送る。ARM7がCC*

CODEC構成レジスタFBUSアドレスマップ

アドレス(31:0)(HEX)	説明
04B0_0000 thru 04BF_FFFF	CODEC0構成レジスタ
04C0_1000 thru 04C0_1FFF	CODEC1構成レジスタ
04C0_2000 thru 04C0_2FFF	CODEC2構成レジスタ
04C0_3000 thru 04C0_3FFF	CODEC3構成レジスタ
04C0_4000 thru 04C0_4FFF	CODEC4構成レジスタ
04C0_5000 thru 04C0_5FFF	CODEC5構成レジスタ
04C0_6000 thru 04C0_6FFF	CODEC6構成レジスタ
04C0_7000 thru 04C0_7FFF	CODEC7構成レジスタ
04C0_8000 thru 04C0_8FFF	予約
04C0_9000 thru 04C0_9FFF	予約

【0110】図53は、カスタマASIC回路網を示したものである。

4. I/Oピン定義

* Uを通して、任意のCODECにある構成データを読もうとする場合、カスタマASICユニットは、アドレスをCODECに送る。カスタマASICユニットは、CODECからのデータを受信すると、トランザクションIXをCCUに返還する。この時点で、構成データがCCUを通してARM7に伝送される。

【0109】

【表35】

※【0111】カスタマASICユニットに対するI/Oピン定義

※【表36】

カスタマASICユニットに対するI/Oピン定義

ピン名	方向	説明
clk1	入力	80MHzシステムクロック入力
f_reset~1	入力	Fbusリセット信号（ローアクティブ）
Fasc_grant_1	入力	ASIFbusをトランクにローアクティブ
Fasc_cs_1	入力	ASICチップセレクト信号（ローアクティブ）
CS_size[7:0]~	入力	CODECデータ伝送サイズ: 8h08=>8バイト、8h18=>24バイト、8h20=>32バイト
DMA-REQ9~	入力	CODECから出るDMAリクエスト信号
CODEC-ACK9~	入力	CODECから出るCODEC確認信号
ref_full	入力	角されるより早くもこの信号のために使
Fdrdy_1	入出力	実際データ以前に1つのサイクル間に有効なFbusデータ準備信号
Fdata[63:0]	入出力	Fbusデータ
Faddr[31:0]	入出力	Fbusアドレス
Freq_ID[9:0]	入出力	FbusリクエストID:[9:6]>=>リクエストID、[5:0]>=>伝送ID
Freq_size[7:0]	入出力	Fbusデータ伝送サイズ
Frdrwr_1	入出力	読出/記入表示: '1'=>読出、'0'=>記入
Fprwr_1	入出力	部分記入表示（ローアクティブ）
CS-DATA[63:8]~	入出力	CODECデータ
Fasc_dfull	出力	ASIFbusに進行FIFOフル。（Fbusに進行）
Fasc_afull	出力	ASIFbusに進行FIFOフル。（Fbusに進行）
Fasc_grCNT[1:0]	出力	サイクログラントの数を表すように必要となるFbusに進行
Fasc_did[2:0]	出力	目的地ICユニットからリクエストにFbus
Fasc_recl_1	出力	ASIFbus信号のトランクにFbus
CODEC[31:0]	出力	CODEC構成とアドレスを中へ記入のため
DMA-ACK9~	出力	DMA確認信号（CODECへ進行される）
Crdrwr_1	出力	CODEC構成レジスタアクセスのために読出/記入表示
CODEC-REQ9~	出力	CODECリクエスト信号
EOP	出力	動作終了。この信号はインタラプト制御器へ進行される
Ref_addr[31:0]	出力	CHANNEL0のためのアドレス（MCUへ進行される）
addr_valid	出力	行される10アドレス有効信号（MCUへ進行される）
Fmcu_grant_1	入力	MCUグラント信号がFbusアービタから出る

【0112】第8章 AD1843 CODECインターフェース

8.1

本章はAD1843 CODECインターフェースに関する説明である。

8.2 概要

AD1843 CODECインターフェースブロックは、AD1843シリアルバーストMSP DMAモジュール間のインターフェースのためのものである。AD1843はシリアルポートを通してデータ及び制御/ステータス情報を送信及び受信する。AD1843は、シ

リアルインターフェースを担当する4個のピン: SDI、SDO、SCLK、SDFSを有する。SDIピンは、AD1843に対するシリアルデータ入力のためのものであり、SDOピンはAD1843からのシリアルデータ出力のためのものである。SCLKピンはシリアルインターフェースクロックのためのものである。AD1843内部と外部の通信において、データビットはSCLKの上昇エッジ以降に伝送され、SCLKの下降エッジでサンプリングされることを要求する。SDFSピンはシリアルインターフェースフレーム同期のためのものである。AD1843 CODECインターフェース

は、マスタモードに基づいたものであって、これはSCLKとSDFS信号がAD1843によって発生されることを意味する。省略時(default)SCLK周波数は、12.288MHzであり、1つのフレームサイクルは48KHzである。CODECインターフェースの基本構造は、DMAに基づいたものである。AD1843インターフェースは、4個の相互に異なるDMAチャンネル：DAC1も対するチャンネル4、DAC2に対するチャンネル5、ADC左側に対するチャンネル6、ADC右側に対するチャンネル7を指定する。DMAからまたはDMAへのチャンネル伝送サイズは、1回当たり64ビットである。従って、DMAチャンネル4とチャンネル5は、2個の相互に異なる32ビットデータ（左側のための16ビットと右側のための16ビット）を送送する。一方、DMAチャンネル6とチャンネル7は、1回に4個の相互に異なる16ビットデータをCODECインターフェースからSDRAMに送る。DAC1とDAC2インターフェースは、各チャンネルのフラグビットが設定された時、データが有効であることを認識する。DAC1とDAC2インターフェースは、フラグビットをチェックしてからDMAを要請する。フラグビットがリセットされると、DAC1とDAC2インターフェースは、DMAリクエストを発生させない。フラグビットの実際の動作は、DMAクロックによって制御される。DMAブロックはフラグビットがリセットされると、DMA認知信号を発生させない。ADC左側及び右側のFIFOが満ちていなければ、DMAリクエスト*

*は発生されない。ソフトウェアはADCフラグレジスタをチェックし、データバスを通して残っているデータを読み出さなければならない。データバスを通してこれらのデータを読み出してから、FIFOは空くようになり、FIFOが満ちるとDMAリクエストを発生させる。AD1843制御レジスタは、制御ワード入力の制御レジスタアドレスと共に読出/記入リクエストを送送することによって、読出及び記入される。読出が要請されると、アドレッシングされた制御レジスタの内容は、次のフレームの間に伝送され、記入が要請されると、記入されるデータはAD1843スロット1に伝送されなければならない。MSPの性能を向上させるために、プログラマーはCODECの制御レジスタを読出または記入する前に、制御フラグレジスタをチェックしなければならない。制御フラグレジスタのフラグビットが設定されると、CODECレジスタの読出及び記入動作が可能である。

【0113】8.3 DMAチャンネル指定

DMAチャンネル4 DAC1左側、右側

DMAチャンネル5 DAC2左側、右側

DMAチャンネル6 ADC左側

DMAチャンネル7 ADC右側

8.4 DMAに対するデータフォーマット

データサイズは64ビットであり、下記のように構成される。

【0114】

	ビット(63:48)	ビット(47:32)	ビット(31:16)	ビット(15:0)
DMA CH4	DA1 LEFT 2ND	DA1 RIGHT 2ND	DA1 LEFT 1ST	DA1 RIGHT 1ST
DMA CH5	DA2 LEFT 2ND	DA2 RIGHT 2ND	DA2 LEFT 1ST	DA2 RIGHT 1ST
DMA CH6	ADC LEFT 4TH	ADC LEFT 3RD	ADC LEFT 2ND	ADC LEFT 1ST
DMA CH7	ADC RIGHT 4TH	ADC RIGHT 3TH	ADC RIGHT 2ND	ADC RIGHT 1ST

【0115】8.5 基本アドレス

04C0_4000 DAC1 BASE

04C0_5000 DAC2 BASE

04C0_6000 ADCL BASE (左側チャンネル)

※04C0_7000 ADCR BASE (右側チャンネル)

8.6 レジスタマップ

【0116】

【表37】

8. 6 レジスタマップ

アドレス	説 明	読出または記入
DACL BASE+0 ADCF BASE+0 ADCR BASE+0	制御レジスタ記入データ入力	W
DACL BASE+2 ADCF BASE+2 ADCR BASE+2	制御ワード入力	W
DACL BASE+0 ADCF BASE+0 ADCR BASE+0	制御レジスタデータ出力	R
ADCL BASE+2	A D C フラグレジスタ	R
ADCL BASE+10	A D C 左側第 1 データ	R
ADCL BASE+12	A D C 左側第 2 データ	R
ADCL BASE+14	A D C 左側第 3 データ	R
ADCL BASE+16	A D C 左側第 4 データ	R
ADCR BASE+10	A D C 右側第 1 データ	R
ADCR BASE+12	A D C 右側第 2 データ	R
ADCR BASE+14	A D C 右側第 3 データ	R
ADCR BASE+16	A D C 右側第 4 データ	R
DACL BASE+20 ADCF BASE+20 ADCR BASE+20	制御フラグレジスタ	R

[0117]

8. 7 レジスタ定義

8. 7. 1 制御レジスタ記入データ入力

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
d15 d14 d13 d12 d11 d10 d9 d8 d7 d6 d5 d4 d3 d2 d1 d0

最上位ビット (MSB) は、伝送された最初のデータ入力ビットである。

8. 7. 2 制御ワード入力

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
r/w ia4 ia3 ia2 ia1 ia0

r/w 読出/記入リクエスト。制御レジスタからの読出または制御レジスタへの記入がフレームごとに発生される。“1”に設定したことは、制御レジスタ読出を示す反面、このビットを“0”にリセットさせることは、制御レジスタ記入を示す。

ia4:0 読出または記入のための制御アドレスレジスタ

8. 7. 3 制御レジスタデータ出力

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
d15 d14 d13 d12 d11 d10 d9 d8 d7 d6 d5 d4 d3 d2 d1 d0

以前フレームでアドレッシングされた制御レジスタの内容

8. 7. 4 ADCフラグレジスタ

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
r4v r3v r2v r1v l4v l3v l2v l1v

r4v-r1v 有効ADC右側データがバッファにある。バッファにあるどのデータが有効であるかを指示する。

l4v-l1v 有効ADC左側データがバッファにある。バッファにあるどのデータが有効であるかを指示する。

8. 7. 5 ADC左側の第1番目のデータ

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
d15 d14 d13 d12 d11 d10 d9 d8 d7 d6 d5 d4 d3 d2 d1 d0

バッファにあるADC左側の第1番目のデータ

8. 7. 6 ADC左側の第2番目のデータ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d15	d14	d13	d12	d11	d10	d9	d8	d7	d6	d5	d3	d3	d2	d1	d0

バッファにあるADC左側の第2番目のデータ

8. 7. 7 ADC左側の第3番目のデータ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d15	d14	d13	d12	d11	d10	d9	d8	d7	d6	d5	d3	d3	d2	d1	d0

バッファにあるADC左側の第3番目のデータ

8. 7. 8 ADC左側の第4番目のデータ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
d15	d14	d13	d12	d11	d10	d9	d8	d7	d6	d5	d3	d3	d2	d1	d0

バッファにあるADC左側の第4番目のデータ

8. 7. 9 制御フラグレジスタ

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
														wfl	rfl

w f l 制御レジスタ記入フラグ。設定されるとCODECは制御レジスタデータを受信する準備をする。

r f l 制御レジスタ読出フラグ。設定されるとCODECは制御レジスタデータを伝送する準備をする。

【0118】第9章 ビデオコーデック

9. 1 概要

ビデオコーデックロジックは、評価(evaluation)ボード上のKS0119とKS0122チップに対しインターフェースし、MSPチップにあるDMAモジュールに対してインターフェースする。KS0119CODECはまたスクリーンリフレッシュ動作を提供する。この動作のために、MCUモジュールに対する直接的なデータ経路は、図54のように具現される。

* 9. 2 上位モジュール定義

上位のモジュールは、図55でのような3個のサブモジュールを有する。

—KS0119スクリーンリフレッシュモジュール

—KS0122ビデオデータキャプチャモジュール

—KS0119及びKS0122チップ構成レジスタをアクセスする、3ワイヤーシリアルホストインターフェースとモジュール。

9. 3 DMAチャンネル指定

DMA CH0	KS0119CODEC
DMA CH1	予約
DMA CH2	KS0122CODEC
DMA CH3	予約
DMA CH4	AD1843オーディオCODEC
DMA CH5	AD1843オーディオCODEC
DMA CH6	AD1843オーディオCODEC
DMA CH7	AD1843オーディオCODEC
DMA CH8	予約
DMA CH9	予約

9. 4. 3ワイヤーホストインターフェースモジュール

このモジュールは、チップ内部のすべてのレジスタが、シリアルインターフェースを通してアクセスされるKS0119とKS0122チップに対してインターフェースする。3ワイヤーシリアルインターフェースモジュール

は、これらのチップに通信プロトコルの機能を支援し、KS0119とKS0122インターフェースロジックのためのレジスタを含む。図3を参照すること。

9. 5 EPROMインターフェース

KS0119 IOピンは、システムがリセットされた後、直ちにプログラムデータをロードすることに使用さ

れ、MSP-1EXブート初期化の一部の、外部EPROMに対するインターフェースとして使用される。より詳細なことはピン指定を参照すること。EPROMはC0000HからDF FFHまでのアドレスでマッピングされたメモリである。

9.6 KS0119レジスタ説明

KS0119は04B0 0000と同一な基本アドレ *

KS0119レジスタアドレスマップ

オフセット (HEX)	レジスタ名
0	フレームサイズレジスタ
1	ID
2	制御/DATA バイト
3	INDEX/DATA0
4	DATA1
5	DATA2
6	DATA3
7	ステータスレジスタ
8	読出DATAシリアルインターフェース
9	読出PROMデータ
A	ロジック制御レジスタ
B	HS、VS極性(Polarity)
C	HSオフセット
D	VSオフセット

[0120] 9.6.2 フレームサイズレジスタ

このレジスタは、図57の図示のとおり、CODECチップに伝送されるフレームサイズを制御し、最小フレームの長さは、3バイトである。

9.6.3 チップIDレジスタ

このレジスタはCODECチップID値を貯蔵するが、KS0119記入に対しては03H、KS0119読出のためには83Hを貯蔵する。

9.6.4 制御/データレジスタ

このレジスタは、次に伝送されるバイトが、レジスタインデックスまたはデータバイトであるという事実を、CODECチップKS0119に伝える。KS0119に対し、08Hは次のバイトがインデックスであることを、09Hは次のバイトがデータであることを表わす。

9.6.5 インデックス/データ0レジスタ

このレジスタは、その以前のバイトに伝送された値によって、CODECチップ構成レジスタに対するインデックス値または、データ0バイトを貯蔵する。プログラミング参照部の通信プロトコルを参照すること。

9.6.6 データ1レジスタ このレジスタは、CODECレジスタIndex+1に記入されるデータを貯蔵する。

9.6.7 データ2レジスタ

このレジスタは、CODECレジスタIndex+2に記入されるデータを貯蔵する。

9.6.8 データ3レジスタ

このレジスタは、CODECレジスタIndex+3に記入されるデータを貯蔵する。

9.6.9 KS0119ロジック制御レジスタ

KS0119制御レジスタに対するビット指定は、図5 50

* スCODEC_REQ0を有し、これは04BF FF FFまで拡張される。

9.6.1 KS0119レジスタアドレスマップ

KS0119レジスタアドレスマップ

[0119]

【表38】

8の図示のとおりである。

9.6.10 HS及びVS極性

このレジスタは、水平同期と垂直同期信号の極性を定義する。0値はアクチブローに定義される一方、1値はアクチブハイに定義される。ビット指定は下記のとおりである。

Bit<0>: VS極性

Bit<1>: HS極性

9.6.11 HSオフセット

アクチブ信号は、このオフセット値以降に発生され、このオフセット値は00Hに定義される。

9.6.12 VSオフセット

アクチブ信号は、このオフセット値以降に発生され、このオフセット値は00Hに定義される。

9.6.13 ステータスレジスタは図59の図示のとおりである。

9.6.14 読出データシリアルインターフェースレジスタ

このレジスタは、読出フラグがビジー(busy)状態から準備(ready)状態への遷移を表してから、シリアルポートからの有効データを貯蔵する。

9.6.15 読出PROMデータレジスタ

このレジスタはPROMフラグが準備状態の場合、有効データを貯蔵する。

9.6.16 プログラミングレ参照

9.6.16.1 構成及び初期化

ビデオディスプレイハードウェアは、2種類のモードすなわち、VGAオーバーレーモードとVGAエミュレーションモードに動作するように製作される。このモード動作は、ロジック制御レジスタにあるビットを設定する

ことにより制御される。

MSSEL:VGAオーバーレーモードの場合0、VGAエミュレーションモードの場合1。

VGAオーバーレーモードでは、P.Cシステム上にVGAカードの存在が要求される。

—モニターケーブルは、MSPカードに連結される。

—支援されるVGA解像度は800×600までである。

ディスプレイバッファは、VGAセッティングと同じサイズであることが要求される。

【0121】ソフトウェアによりVGAフレームバッファで、カラーキー四角領域が満たされるビデオウィンドーを設定するために、ビデオデータはMSP SDRAMでVGAフレームバッファにある四角領域と同一サイズと位置の四角領域に記入されなければならない。図60を参照すること。KS0119チップはカラーキーを認識し、VGA入力ポートをビデオ入力ポートに切り換える。ソフトウェアによりDMAチャンネル0スタートアドレスを、SDRAMビデオ出力バッファ上位の左側に設定し、DMAレコードの長さは、VGAカードに設定された解像度とビデオデータで使された画素当りのビット(4:2:2=画素当たり16ビット)によって設定される。

9. 6. 16. 2 KS0119に対するシリアルプロトコル3-ワイヤーインターフェース

KS0119チップにある構成レジスタを設定する場合、プロトコルは下記のとおりである。

—周辺チップに伝送されるためには、最少に2個のフレームが必要である。

—第1番目のフレームは、構成レジスタのインデックスを設定するためのものである。

—第2番目のフレームは、データ(レジスタの内容)の読出または記入のためのものである。

ソフトウェアによりフレームサイズレジスタを適切な長さで設定し、シリアルアクセスビットを1に設定する。そうすると、フレームサイズレジスタを変更する前、フレームに必要なすべてのバイトをソフトウェアによりロードし、CODECインターフェースロジックは、フレームシリアル化が開始される前、すべてのバイトがロー
ドされる時まで待つ。第1番目に伝送されるフレーム *

* は、インデックスを設定するためのもので、フレームサイズは3である。図61を参照すること。

【0122】第2番目のフレームは、レジスタを設定するためのもので、フレームサイズは3である。各データバイトの以降、チップはインデックスを1ずつ自動に増加させ、これは複数バイトのデータを4個データバイトまで支援するCODECインターフェースロジックに伝送することによって、連続的なレジスタを設定することを可能にする。読出または記入動作が遂行された場合、

10 ソフトウェアにより読出動作時に、有効データのためのステータスレジスタの読出及び記入フラグをチェックするか、次のフレームを伝送する前、記入フラグ=準備(ready)であるかをチェックする。次の例は、KS0119データシートを設定する段階を示している。2個のレジスタが連続的なインデックスを有するので、この2バイトは単一フレームにロードされ得る。まず、インデックスは下記のとおり設定されなければならない。

—8H値(フレームサイズ=3、シリアルアクセスビット設定)を有するロードフレームサイズレジスタ(Address=04B0_0000H)

20 —03値を有するロードIDレジスタ(Address=04B0_0001H)

—ロードデータ/制御バイト:KS0119に、次のバイトがインデックスであることを知らせる08H値(Address=04B0_0002H)

—6H値を有するロードインデックスレジスタ(Address=04B0_0003H)

シリアルインターフェースは、フレームサイズレジスタにある内容の一致可否を検出しフレーム伝送を開始し、ステータスレジスタにある記入フラグは、ビジー(busy)状態に設定される。次のフレームを伝送する前、ソフトウェアによりステータスレジスタにあるフラグをチェックする。フラグが準備状態であれば、ソフトウェアにより次のフレームのための値をロードすることができる。

9. 7 KS0122レジスタ説明

KS0122は04C0 2000に該当する基本アドレスを有し、これは0420 2FFFまで拡張される。

9. 7. 1 KS0122レジスタアドレスマップ

【0123】

【表39】

9. 7. 1 KS0122レジスタアドレスマップ

アドレス (HEX)	レジスタ名
0	フレームサイズレジスタ
1	ID
2	制御/DATAバイト
3	IDEX/DATA0
4	DATA1
5	DATA2
6	DATA3
7	予約
8	読出DATAシリアルインターフェース
9	予約
A	ロジック制御レジスタ
B	予約
C	予約
D	予約
E	ステータスレジスタ

【0124】9. 7. 2 フレームサイズレジスタ

このレジスタは、図62に定義されたように、CODECチップに伝送されるフレームサイズを制御し、最小フレームの長さは3バイトである。

9. 7. 3 チップIDレジスタ

このレジスタはCODECチップID値を貯蔵するが、KS0122記入に対しては04H、KS0122読出 20のためには84Hを貯蔵する。

9. 7. 4 制御/データレジスタ

このレジスタは、次に伝送されるバイトが、レジスタインデックスまたはデータバイトであるとの事実をCODECチップKS0122に伝える。KS0122に対し、00Hは次のバイトがインデックスであることを、01Hは次のバイトがデータであることを表わす。

9. 7. 5 インデックス/データ0レジスタ

このレジスタは、その以前のバイトに伝送された値によって、CODECチップ構成レジスタに対するインデックス値或いはデータ0バイトを貯蔵する。プログラミング参照部の通信プロトコルを参照すること。 30

9. 7. 6 データ1レジスタ

このレジスタは、CODECレジスタIndex+1に記入されるデータを貯蔵する。

9. 7. 7 データ2レジスタ

このレジスタは、CODECレジスタIndex+2に記入されるデータを貯蔵する。

9. 7. 8 データ3レジスタ

このレジスタは、CODECレジスタIndex+3に 40記入されるデータを貯蔵する。

【0125】9. 7. 9 KS0122ロジック制御レジスタ

KS0122制御レジスタに対するビット指定は、下記のとおりである。

bits<1:0>

00 4:2:2フォーマット

01 4:1:1フォーマット

10 CCIR656フォーマット

9. 7. 10 ステータスレジスタ

ビット<0>: フィールドステータス

0: 偶数フィールド

1: 奇数フィールド

ビット<1>: VSステータス

0: 1から0までのVS

1: 0から1までのVS

9. 7. 11 読出データシリアルインターフェースレジスタ

このレジスタは、読出フラグがビジー(busy)状態から準備(ready)状態への遷移を表してから、シリアルポートからの有効データを貯蔵する。

9. 7. 12 KS0122に対するシリアルプロトコル3-ワイヤーインターフェース

KS0122チップにある構成レジスタを設定する場合、プロトコルは下記のとおりである。

—周辺チップに伝送されるためには、最少2個のフレームが必要である。

—第1番目のフレームは、構成レジスタのインデックスを設定するためのものである。

—第2番目のフレームは、データ(レジスタの内容)の読出または記入のためのものである。

【0126】ソフトウェアによりフレームサイズレジスタを適切な長さに設定し、シリアルアクセスビットを1に設定する。そうすると、フレームサイズレジスタを変更する前、フレームに必要なすべてのバイトをソフトウェアによりロードし、CODECインターフェースロジックは、フレームシリアル化が開始される前、すべてのバイトがロードされる時まで待つ。第1番目に伝送されるフレームはインデックスを設定するためのもので、フレームサイズは3である。図63を参照すること。第2番目のフレームはレジスタを設定するためのもので、フレームサイズは3である。各データバイトの後、チップはインデックスを1ずつ自動に増加させ、これは複数バイトのデータを4個データバイトまで支援するCODECインターフェースロジックに伝送することによって、連続的なレジスタを設定することが可能にする。読出または記入動作が遂行された場合、ソフトウェアにより読

出動作時に、有効データのためのステータスレジスタの読出及び記入フラグをチェックするか、次のフレームを伝送する前、記入フラグ=準備(ready)であるかをチェックする。次の例は、KS0122データシートを設定する段階を示している。クロマキーバイト0とバイト1に対する値を設定するために、このレジスタのためのインデックスは、バイト0に対し6AH、バイト1に対し6BHである。KS0122データシートを参照すること。2個のレジスタが連続的なインデックスを有するので、この二バイトは単一フレームにロードされ得る。ま

【0127】-83H値(フレームサイズ=3、シリアルアクセスビット設定)を有するロードフレームサイズレジスタ(Address=04B0_0000H)

-03値を有するロードIDレジスタ(Address=04B0_001H)

-ロードデータ/制御バイト:KS0122に、次のパ*

10.2 略語

A/V	オーディオ及びビデオ
BP	ビットストリーム処理器(MSPブロック)
CCU	キャッシュ制御ユニット(MSPブロック)
CIF	29.97Hzで352×288の輝度サンプル解像度を有する、共通中間フォーマット
DCT	離散余弦変換
DMA	直接メモリアクセス
DSM	デジタル貯蔵メディア
FBUS	速いバス(MSP内部データバス)
GOB	ブロックグループ
GSTN	一般スイッチテレフォンネットワーク(既に公知されたPSTN)
HDD	ハードディスクドライバ
I/F	インターフェース
IOBUS	入出力バス(MSP内部周辺バス)

ITU-T-601 29.97Hzで720×480と、25Hzで720×576のそれぞれのサンプル解像度を有する、カラーテレビジョン信号のデジタルコーディング用のテーブル基準(以前はCCIR601とも呼ばれる)。しかし、ディスプレイ解像度は、720×480または704×480である場合もある。

LSB 最少ビット

LUT ルックアップテーブル

MPEG モーション映像専門家(expert)グループ

MSB 最大有意ビット

MSP 三星マルチメディア信号処理器

QCIF 29.97Hzで176×144の輝度を有するQuarter_CIF

RLC RUN_長さ及びレベルコード

SDRAM 同期ダイナミックランダムアクセスメモリ

SIF NTSC用29.97Hzで352×24

* イトがインデックスであることを知らせる08H値(Address=04B0_0002H)

-6H値を有するロードインデックスレジスタ(Address=04B0_0003H)

シリアルインターフェースは、フレームサイズレジスタにある内容の一致可否を検出してフレーム伝送を開始し、ステータスレジスタにある記入フラグはビジー(busy)状態に設定される。次のフレームを伝送する前、ソフトウェアによりステータスレジスタにあるフラグをチェックする。フラグが準備状態であれば、ソフトウェアにより次のフレームのための値をロードすることができる。

【0128】第10章 ビットストリーム処理器 10.1

本章は、ビデオデータ圧縮及び伸長応用のための主要MSP処理エンジン中の1つのビットストリーム処理器(BP)を設計するための、機能的な要求条件を説明する。

0及び、PAL用の25Hzで352×288の輝度解像度を有する、MPEG-1ビデオテーブル基準用の情報入力及びフォーマット

TSD 定義される

VLC 可変長さコード

40 VP ベクトルプロセッサ(MSPブロック)

10.3 主要特徴

MPEG-1、MPEG-2、H.261及びH.263のエンコーディング及びデコーディング応用とスライス(またはGOB)層をなして解釈する構文を支援する。

*実時間でRLC処理を遂行

*MPEG-1、MPEG-2、H.261及びH.263ビデオ標準にあるすべてのハフマンテーブルを用いて実時間でハフマンコード処理を遂行。

*2個の順方向/逆方向ジグザグスキャン変換方式を支

援。

* 731.4Mbits/sec (32-bit@40MHz) の最大伝送レートでインターフェースするIOBUS

* 最大動作クロック周波数は40MHz

* ハフマンコードックアップテーブルのための9.2Kbit ROMを含む。

* 320byte内部SDRAMを含む。

* 先占(pre-emptive)及び協力文脈スイッチングモードを支援

* 制御経路のための目的ゲート計算は、6K gates + RAM及びROM

【0129】10.4 概要

ビットストリーム処理器(BP)は、4個のMSP内部周辺装置の中の1つである。これは、ビデオ圧縮及び復元状態の数ビットストリームを支援するために、ハードウェア組織ブロックである。このような装置は特に、MSP内部のVP及びARM7がこのようなビットの操作に効率的なアーキテクチャーを有していないので、ビットレベル処理のために設計された。このようなBPは、731.4Mbits/secの最大伝送速度を有するIOBUSと呼ばれる32ビットバスを通してデータを送受信する。そしてBPは、独立的な処理装置として動作し、ARM7またはVPのソフトウェアにより制御される。その上、特にBPはスライスまたはGOB及び、その以下に含まれているすべての情報をエンコーディング及びデコーディングし、そしてCCUから／にデータを送受信する。前記のBPはまた、順方向及び逆方向ジグザグ変換を遂行し、差動DC係数をエンコーディング及びデコーディングする。さらに、このようなBPはデコーディングで差動モーションベクトルを使用してモーションベクトルを復元し、2個の特殊なモードすなわち、MPEG-2エンコーディングでデュアルプライムモードと、H.263エンコーディング及びデコーディングで予測モードを除いて、エンコーディングではその反対の動作を遂行する。もし、BPが単純なモード*

*で動作すると仮定すると、BPは一応スライスまたはGOBを処理しつつ始め、BPはスライスまたはGOB処理が完了してからインタラプトされる。このような動作は、全二重モードがスライスまたはGOBをインタリーブによりエンコーディング及びデコーディングすることにより遂行される。もし、ARM7がBPを他の作業に瞬間的にスイッチングさせることを願うと、BPは現在のスライスまたはGOBが完了される前、BP過程を完了する先占文脈切換モードを支持するようになる。

【0130】図3はBPのブロックダイアグラムを示したものである。図3の図示のとおり、BPは5個のブロックIOBUSインターフェース装置、VLC FIFO装置、VLC LUT ROM、制御状態マシン及びBPコア装置とを含む。入出力データは、16×32ビットラムを含む、IOBUSインターフェース装置により動作される。これはすべてのデータ移動及びインタラプト要求を支援する。VLC FIFO装置は、データ復号化動作のために、次のデータワードを準備し、そしてデータ符号化動作のために、出力データバッキングを遂行する。VLCルックアップテーブルROMは、すべてのハフマンコードの処理のために、すべての必要な情報を貯蔵する768×12bitのサイズを有する。制御状態マシンを設計する時、すべてのエンコーディング及びデコーディングを制御する。BPコア装置は加算器、比較器、パレルシフター、レジスタファイル及び、128×16ビットのRAMを含む小さいプロセッサである。ビット操作は、前記のコアに有用である。

【0131】10.5 信号定義

BP外部インターフェースに要求される信号は、表45に示している。文字“1”の最後にある信号は、アクティブローを示す。テーブル1の“方向”コラムで“B”、“I”“O”は両方向信号であって、入力信号及び出力信号とをそれぞれ意味する。

【0132】BP信号定義

【表40】

B P 信号定義

信 号	サイズ	方向	説 明
IOBUS[31:0]	32	B	32ビット両方向とマルチプレク スされた動作が完了した場合に、 スレーブデバイスとして の信号を出力する。
IOB_rd_l	1	B	すべての出力／記入伝送のための、 マスタ／スレーブ動作が完了した 場合、アクティブステートで出力す る。
IOB_wr_l	1	B	すべての出力／記入伝送のための、 マスタ／スレーブ動作が完了した 場合、アクティブステートで出力す る。
IOB_ready_l	1	B	データがスレーブモードで有効に読 み出される時にマスタがアクティブ ステートで出力する。
IOB_cs_bsp_l	1	I	一応アドレスがラッチされてからリ クエストを処理するまでの、アー ビタからBPへのアクティブ信号
IOB_ale_l	1	I	マスタ／スレーブ動作のための、ア ービタの出力でBPへの入力である アクティブアドレスストローブ
IOB_req_bsp_l	1	O	バス読出及び記入伝送のマスタとな るための、BPからアービタへのア クティブリクエスト
IOB_tsize[1:0]	2	B	4、8、12及び16バイトの中の いずれか1つと同一な伝送サイズを 表示するための、マスタからスレー ブへ入力されるデータ
IOB_grant_bsp_l	1	I	アービタからBPへのアクティブロー グランド信号
Reset_l	1	I	アクティブローシステムリセット
Clk40M	1	I	40MHzクロック
ARM7_IRQ	1	O	ARM7により解除されなければなら ないARM7用の、BPから実行され ないラプトルクエスリクエスリ 、前記のBPによリクエスリクエ 、前記のBPによリクエスリクエ 、前記のBPによリクエスリクエ ントローラーに伝送される。

【0133】10.6 エンコーディング／出コーディ
ング用のデータ流れ図

ここで例えば、代表的なビデオエンコーディング及びデ
コーディング応用のデータ流れを含む。ここでは、オー
ディオデータの流れについては詳細に記述しない。

10.6.1 エンコーディングの場合

段階E1：ロー (RAW/Vデータ入力)

普通の入力ビデオ及びオーディオ信号がサンプリングさ
れ、外部コーデックによりデジタル化され、そして使用
者ASICに供給される。しかし、マルチメディアPC
環境で、あるVGA制御ボードはまたフレーム捕獲子(g
rabber)とサウンドキャプチャを含む。従って、ロー
(RAW)A/Vデータは、使用者ASICまたはPCIバ
スインターフェースの中のいずれか1つから伝達され
る。カスタムASICまたはPCIバスは、32BYTES
の小さいバッファを含む。このバッファにあるデー

タはDAM操作を利用したFBUSを通して、外部SD
RAMに伝達される。このようなデータ移動は、電源が
リセットされてからARM7により初期化される。

段階E2：VPによるプリフィルタリング

まず、VPはVPデータ開始（一般的にスクラッチパッ
ド領域）のSDRAMに貯蔵されたイメージデータをフ
ェッチする。そして、VPはこのような画素を一時的に
フィルタリングしてから空間をスケーリングする。プリ
フィルタリングしてから、映像の解像度は正常的にIT
U_T_601サイズからCIFまたはQCIFサイズ
に変換される。このVPは、外部SDRAMに対するプ
リフィルタされた結果を記録する。

【0134】段階E3：VPによるデータ圧縮

VPは、これに対応する標準に提示された法則によつて
圧縮が遂行されるように、さらにVPデータキャッシュ
の中にSDRAMのプリフィルタされたデータをフェ

ッチする。正常的にVPは、順方向DCT/順方向適応量子化、モーション予測、マクロブロックタイプ決定などを遂行する。このような過程を遂行してから、VPは更にVPデータキャッシュへ適当なヘッド情報を有する結果を記録しなければならない。实际的に、このVPデータキャッシュ領域は、BP入力バッファとして利用される。バッファの状態を検査するために、フラグ信号が利用される。

段階E4：ARM7によるBP初期化

实际的にBPが動作される前、ARM7はBPの初期レジスタを初期化しなければならない。このような初期化は、パワーオンリセット信号が印加されてから、128サイクルの間には遂行されない。特にARM7は、入力バッファアドレス及びBP命令レジスタを初期化させなければならないし、スライスまたはGOB内で符号化されたマクロブロック数を指定しなければならない。このようなレジスタを初期化してから、ARM7はBP過程を遂行するようにBPイネーブルフラグをセットしなければならない。

段階E5：BPによりビットストリーム過程

もし、入力2個のバッファの中のいずれか1個がフル(full)の場合、BPはIOEJSを通してデータを読み込み始める。すなわちBPは、バッファがフルの場合のみデータを読み込める。そして、BPはジグザグフォーマットで8×8ブロックデータを変換させる。そして、その結果は直接にRLC及びハフマン符号化される。このようなハフマン符号化された結果は、ARM7データキャッシュまたはSDRAM中の中のいずれか1つに伝送され得る。BPは、前記のバッファがオーバーフローされないように空いている場合のみ、出力バッファに書き込むべきである。この過程の最後の例を挙げれば、処理されたマクロブロック数がARM7により指定されたマクロブロック数と同一な場合、BPは最後のデータのバイト及び位置でARM7とインタラプトするようになり、現在スライスまたはGOB過程を終了する。

【0135】段階E6：ARM7によりビットストリーム形成とA/Vマルチプレキシング

ARM7はハフマン符号化されたデータ及び構文パラメータを結合し、最後のビットストリームを作り、その過程を反復する。そしてARM7はまた、スライスまたはGOBマルチプレクスオーディオ及びビデオビットストリームの上部層と操作することができる。この結果は、ARM7によりSDRAMに書かれる。

段階E7：VPによるネットワークインターフェース (ビデオ会議用選択)

ビデオ電話またはビデオ画像会議の応用のために、前記の段階6まではVPがH.324GSTNビデオ電話用のV.34モデムまたは、H.320ISDNビデオ会議端末用の1400系列のインターフェースのようなネットワークインターフェースがその機能を遂行してきた。

段階E8：最後のビットストリーム出力

SDRAMに貯蔵された最後のビットストリームは、カスタASICまたはPCIの中のいずれか一つに伝送される。正常的に使用者ASICブロックはネットワークインターフェースに使用され、そしてPCIバスインターフェースは、記録装置(例えば、HDD)データ貯蔵のために利用される。このデータが移動する時には、ARM7により初期化されたDMAデータ伝送を利用する。

【0136】10.6.2 デコーディングの場合

段階D1：ビットストリームフェッチ

マルチメディア環境で圧縮されたビットストリームは、CD-ROMドライバー、HDD及びネットワークインターフェースの中のいずれか一つから供給される。従って、このビットストリームは、カスタASICまたはPCIバスの中のいずれか1つとなる。カスタASICまたはPCIバスの32byteに貯蔵されたデータは、DMAを利用したSDRAMに伝送される。

段階D2：VPによるネットワークインターフェース

(ビデオ会議用選択)

ビデオ会議において、データはまず、VPによりV.34または1400系列のネットワークインターフェースルーチンが遂行される。VPはSDRAMに対する結果を書き込む。

段階D3：ARM7によりA/Vディマルチプレキシング及びヘッダ分析

ARM7はSDRAM内のデータをARM7データキャッシュに移動させ、A/Vビットストリームディマルチプレキシングを遂行する。ビデオビットストリームのためにARM7はまた、すべてのスタートコードを検索し、そしてスライスGOBが検出するまでヘッダを分析する。ARM7は復号化されたビットストリーム構文パラメータを、ARM7によりSDRMの特別領域に貯蔵させる。ディマルチプレキシングされたオーディオ及びビデオビットストリームは、SDRAMにあるレートバッファにそれぞれ伝送される。各動作のために、レートバッファのサイズを異にしてもよい。例えば、ビデオ速度バッファサイズのために、MPEG-1は370Kbitsに、MPEG-2は1.835Mbitsに勧告する。

【0137】段階D4：ARM7によるBP初期化

この段階の遂行は、ただ符号化されたマクロブロック数に対してレジスタの初期化を要求しないことを除いては、以前のサブセクションの段階E4と同様である。すなわち、初期化はパワーオンリセット信号が印加されてから、128サイクルの間に遂行されてはいけない。

段階D5：BPによるビットストリーム過程

特別なスライスまたはGOBのためにBPを初期化してから、復元されたデータは2個のバッファに伝送する。BPはフルフラグの状態を検査するIOBUSを

10

20

30

40

50

通してデータを読み込む。BPはもし、入力データがヘッドワードを含んでいると、構文パラメータを分析する。もし、BPが続く次のビットをハフマンコードで認識すると、各ハフマンコード用の最上4サイクル以内にハフマンデコーディングを遂行する。もし、ハフマンデコードがDCT AC係数であれば、ハフマンデコードされた結果が、64画素成分を表わすデコードされたRLCとなる。再現画素はこれとは反対に、ジグザグに変換され、そして最後にVPが順方向量子化を遂行するように、2個の出力バッファに伝送される。BPはスライスまたはGOBでない初期コードを検出してから、このような過程を遂行し続ける。もし、これが検出されなければ、BPは最後に使用されたデータに対して、バイト及びビットの位置情報を有するARM7とインタラプトさせる。そうすると、ARM7は次のスライスまたはGOBスタートコードを検索し、このような過程を繰り返す。

【0138】段階D6：VPのデータ復元

段階D5の結果を使用し、VPは逆量子化、逆DCT及びモーションベクトルを利用した映像再現を遂行する。符号化過程を完了してから、VPはSDRAMの中にある結果を貯蔵する。

段階D7：VPの以降の過程

ビデオ及びオーディオデータが、デジタル／アナログ変換器に伝送される前、画素はVPが好ましい出力解像度及びイメージを得るように、前記の過程を遂行する。こ*

BP内部レジスタ

アドレス (HEX)	レジスタ名	サイズ	説明
0_0000	BP_MODE[31:0]	32	BP処理モードレジスタ
0_0004	BP_CONTROL[31:0]	32	BP制御レジスタ
0_0008	IBUF0_START[31:0]	32	入力バッファ0のスタートアド
0_000C	IBUF0_END[31:0]	32	入力バッファ0の終了アドレス
0_0010	IBUF1_START[31:0]	32	入力バッファ1のスタートアド
0_0014	IBUF1_END[31:0]	32	入力バッファ1の終了アドレス
0_0018	OBUF0_START[31:0]	32	出力バッファ0のスタートアド
0_001C	OBUF0_END[31:0]	32	出力バッファ0の終了アドレス
0_0020	OBUF1_START[31:0]	32	出力バッファ1のスタートアド
0_0024	OBUF1_END[31:0]	32	出力バッファ1の終了アドレス
0_0028	SAVE_ADR[31:0]	32	文脈貯蔵スタートアドレス
0_002C	VALID_BYTE_ADR[31:0]	32	ライン入力または出力ダブルバッファで使われる最新データのバイトアドレス
0_0030 0_0031 0_004F	BP_STATUS[0] BP_STATUS[1] BP_STATUS[30]	1 — 1	BPステータスレジスタのLSB 第2LSB 第2MSB
0_0050	BP_STATUS[31:0]	32	BPステータスレジスタ
0_0054	BP_INT_MASK[15:0] V_MB_SIZE[7:0] H_MB_SIZE[7:0]	32	BPインタラプトマスクレジスタ及び、多数のマクロブロック内の画像垂直及び水平サイズ
0_0058	ARM7_IRQ[0]	1	ARM7インタラプトトリグレス
0_005F ~ 0_006F ~ 0_007F ~	— BP_CACHE[Address] —	— 8 —	未来拡張用に予約される。 BPキャッシュ領域 未来拡張用に予約される。

【0141】*BP-MODE[31:0] (読出専

*のような結果はまた、SDRAMに貯蔵される。

段階D8：ロー(RAW)A/Vデータ出力

最後に、SDRAM内部の再現オーディオ及びビデオデータは、DMAを利用して出力される。さらに、このようなデータ移動はARM7により初期化される。現在のビデオオーバーレー技術は、PCIバスがビデオソースにデータを伝送できるようにし、最後にデータはカスタマASICまたはPCIバスの中のいずれか1つに伝送される。

【0139】10.7 プログラミングモデル

10.7.1 BPベース装置アドレス

BPは次の32ビット基本装置アドレスを有している。〈MSP_BASE〉〈BP_BASE〉〈Address_Offset〉ここで、〈MSP_BASE〉はMSPベースPCI装置アドレスにより規定された5ビットであり、〈BP_BASE〉は7'b1111100に等しい7ビットであり、〈Address_Offset〉はBP内部レジスタに割当てられた20ビットである。従って、全体MSP I/O装置アドレスマップで、BPに割当てられたアドレス範囲は、27'h7C0_0000から27'h7CF_FFFFまでである。

10.7.2 内部レジスタ説明

内部レジスタセットは、表に示しており、表のすべてのレジスタは、ARM7またはVPにより書かれるか読まれることができる。

【0140】BP内部レジスタ

【表41】

用、省略時の値なし) — このレジスタはビデオ標準タイ

ブと多様な画像レベル情報を定義し、詳細なことはサブセクション10. 8. 1で示している。

*BP_CONTROL [31:0] (読み取り書き込み、省略時の値は“32'h 0000_0000”) - このレジスタは、BP動作のために多様な制御パラメータを含む。ARM7またはVPは、このレジスタにある各フラグをセットし、あるフラグはBPによりセットされる。ビット仕様はサブセクション10. 8. 2で示している。

*IBUF0_START [31:0] (読み取り書き込み、省略時の値なし) - このレジスタは、BP入力両方向バッファの入力バッファ0となるように、初期アドレスをARM7により定義し初期化する。IBUF0_START用の初期化値は常にIBUF0_ENDより小さく、IBUF0_START [3:0] は4'b0000と同一である。

*IBUF0_END [31:0] (読出専用、省略時の値なし) - このレジスタはBP入力両方向バッファの入力バッファ0で最後のアドレスを定義しており、この内容はセクション10. 11に記述されている。

*IBUF1_START [31:0] (読み取り書き込み、省略時の値なし) - このレジスタはBP入力ダブルバッファの入力バッファが1となるように、ARM7のスタートアドレスを初期化させる。IBUF1_STARTの初期化値は常にIBUF1_ENDより小さく、IBUF1_START [3:0] は4'b0000と同一になる。この内容はセクション10. 11に記述されている。

*IBUF1_END [31:0] (読出専用、省略時の値なし) - このレジスタはBP入力ダブルバッファの入力バッファ1が1となるように、最後のアドレスを定義する。この内容はセクション10. 11に記述されている。

*OBUF0_START [31:0] (読み取り書き込み、省略時の値なし) - このレジスタはBP出力ダブルバッファの出力バッファが0となるように、ARM7のスタートアドレスを初期化させる。OBUF0_STARTの初期化値は、OBUF0_ENDより常に小さく、OBUF0_START [3:0] は4'b0000と同一である。この内容はセクション10. 11に記述されている。

*OBUF0_END [31:0] (読出専用、省略時の値なし) - このレジスタはBP出力ダブルバッファの出力バッファが0となるように、最後のアドレスを定義する。この内容はセクション10. 11に記述されている。

*OBUF1_START [31:0] (読み取り書き込み、省略時の値なし) - このレジスタはARM7によりBP出力ダブルバッファの出力バッファが1となるように、ARM7のスタートアドレスを初期化する。OBUF1_STARTの初期化値は、OBUF1_ENDより常に小さく、OBUF1_START [3:0] は

4'b0000と同一である。この内容はセクション10. 11に記述されている。

*OBUF1_END [31:0] (読出専用、省略時の値なし) - このレジスタはBP出力ダブルバッファの出力バッファが1となるように、最後のアドレスを定義する。この内容はセクション10. 11に記述されている。

*SAVE_ADR [31:0] (読出専用、省略時の値なし) - このレジスタは先占文脈切換モードが要求される場合、BP内部文脈を貯蔵するように、SDRAMの初期アドレスに定義する。関連資料はサブセクション10. 12. 1を参照する。

[0142] *VALID_BYTE_ADR [31:0] (読み取り書き込み、省略時の値なし) - このレジスタはデコーディングで入力ダブルバッファまたは、エンコーディングで出力ダブルバッファの最後の有効データバイト位置を表わす。このレジスタの目的は、ARM7及びBPの間でハンドシェーキングするためのものである。一般に、有効バイトデータの有効ビット位置のために追加的な情報が要求されるが、これはBP_CONTROL [31:0] レジスタ内に含まれている。詳細な内容はセクション10. 13にある。

*BP_STATUS [31:0] (読み取り書き込み、省略時の値は“32'h 0000_0000”) - このレジスタは、BPの多様な内部状態を表わす。最下2バイト (例えば、BP_STATUS [15:0]) のすべてのビット位置は、ARM7_IRQを“1”にセットすることができるインタラプト条件である。このレジスタは2つの方法で接近することができる。ARM7またはVPアドレス27'h7C0_0050を使用する全32-ビットレジスタを読取または書込可能である。しかし、一般的にARM7及びVPは、ビット単位でBP_STATUSレジスタの内容を書込 (またはリセット) することが好ましい。BPはまた、BP_STATUSの各ビット当り27'h7C0_0030から27'h7C0_004Fまでの範囲のアドレスを割当てることによって、この特徴的な内容を支援する。このようなビット内容は、サブセクション10. 8. 3に記述されている。

*BP_INT_MASK [15:0] (読出専用、省略時の値は“16hFFFF”) - このレジスタの各ビットは、前記のBP_STATUS [15:0] によるインタラプト条件に対応し、BP_STATUS [15:0] の内部でコーディングされる前の条件を有する論理的な値 (and-ed) である。もし1つのマスクビットが“0”にセットされると、対応インタラプト条件は、無条件的に“0” (例えば、ディスエーブルされる) にセットする。このようなインタラプトに対する詳細な内容は、セクション10. 9に記述されている。

*V_MB_SIZE [7:0] (読出専用、省略時の値なし) - このレジスタは符号化または復号化される画

像の垂直サイズを表わす。ここで、この値はマクロブロック数を意味する。例えば、もし垂直サイズが288画素であれば、 $V_MB_SIZE[7:0] = 288 / 16 = 18$ となる。ARM7はBPエンコーディング及びデコーディング動作をスタートする前に常に設定しなければならない。

【0143】*H_MB_SIZE[7:0] (読出専用、省略時の値なし) - このレジスタは符号化または復号化される画像の水平サイズを表わす。ここで、この値はマクロブロック数を意味する。例えば、もし垂直サイズが352画素であれば、 $H_MB_SIZE[7:0] = 352 / 16 = 22$ となる。ARM7はBPエンコーディング及びデコーディング動作をスタートする前に常に設定しなければならない。

ARM7_IIRQ[0] (読出専用、省略時の値は“0”) - このレジスタは、ARM7にインタラプトを

* 要求するための1ビットフラグであり、ARM7_IIRQ出力ポートに直接連結されている。もしBP_STAT[15:0]の任意ビットが“1”にセットされると、このフラグはセットされる。そしてARM7はこのフラグをリセットさせる。

10.8 BP I/Oデータワードフォーマット

このセクションでは、BP入出力用命令語データ及びマクロブロックデータワードフォーマットを含む。

10.8.1 BP_MODEレジスタフォーマット

27'h7C0_0000アドレスの32ビットBP_MODEレジスタは、テーブル25に与えられた下記のフォーマットを有している。すなわち、BP_MODE[31] = PARAM_SET2[7]とBP_MODE[0] = SF[0]とを表わす。

【0144】

【表42】

BP_MODEレジスタフォーマット

バイト/ビット	7	6	5	4	3	2	1	0
バイト0	PS		PT		—	SF		
バイト1	PARAM_SET0							
バイト2	PARAM_SET1							
バイト3	PARAM_SET2							

【0145】*standard_format[S F] - 使用されるビデオ標準は、テーブル26に定義されている。前記のSFは、常にBPがすべてのビデオエンコーディング及びデコーディング応用にイネーブルさ*

※れる前に、ARM7により定義されなければならない。

【0146】

【表43】

SF定義

バイト [2:0]	標準フォーマット
3'b000	MPEG-1ビデオエンコーディング
3'b001	MPEG-1ビデオデコーディング
3'b010	MPEG-2ビデオエンコーディング
3'b011	MPEG-2ビデオデコーディング
3'b100	H. 261エンコーディング
3'b101	H. 261デコーディング
3'b110	H. 263エンコーディング
3'b111	H. 263デコーディング

【0147】*picture_type (PT) - 映像コーディングタイプは、テーブル27に定義されている。PT用値00は、MPEG-1、MPEG-2及びH. 263応用のための特殊な場合である。特にD_映像は、たとえMPEG-2に使用されなくても、MPE*

*G-2用の映像タイプに割当てられる。その理由は、MPEG-1ビットストリームがMPEG-2ビットストリームのサブセットからである。

【0148】

【表44】

PTの定義

バイト0[5:4]	MPEG-1	MPEG-2	H. 261	H. 263
2'b00	D-画像	D-画像	不適合	PB-フレーム
2'b01	イントラ	イントラ	不適合	イントラ
2'b10	予測化	予測化	常に	インター
2'b11	両方向	両方向	不適合	不適合

【0149】*picture_structure ※する。

(PS) - 映像構造情報は、表43に定義されている。 【0150】

更に、PS用値00は非論理的であるのでエラーを招来※40 【表45】

PSの定義

バイト0[7:6]	MPEG-1	MPEG-2	H. 261	H. 263
2'b00	不適合	不適合	不適合	不適合
2'b01	不適合	上位フィールド	不適合	不適合
2'b10	不適合	下位フィールド	不適合	不適合
2'b11	フレーム	フレーム	フレーム	フレーム

【0151】*parameter_set0、1及び2 (PARAM_SET0、PARAM_SET1、PARAM_SET2) — このよう
な3バイトは、MPEG-1、MPEG-2及びH. 263に使用される多様なパラメータで定義される。各パ*

*ラメータセット用定義は、表に記述されている。

【0152】

【表46】

PARAM_SET0の定義

バイト/ビット	7	6	5	4	3	2	1	0
バイト1	VSF CPM	AS	IVF	CMV AP	FPFD	TFF	IDP	

【0153】*intra_dc_precision (IDP) — MPEG-2に定義された2ビットイントラdc 精度パラメータは、MPEG-1応用で00にセットされなければならない。

*top_field_first (TFF) — モーションベクトルエンコーディング及びデコーディングに使用されるMPEG-2用フラグである。

*frame_pred_frame_dct (FPFD) — MPEG-2用フラグは、フレームDCT及びフレーム予測が使用されることを表わす。

*concealment_motion_vectors (CMV) またはadvanced_prediction_mode (AP) — MPEG-2で、このフラグはモーションベクトルが映像間のマクロブロックで使用されることを表わす。H. 263で、このフラグはもし、改良予測モードがONであれば、1にセットされる。そうでない場合は0にセットされる。次の標準のため
めにこのフラグは0にセットされなければならない。 ※

※*intra_vlc_format (IVF) — MPEG-2用フラグは、映像間のマクロブロックのためのVLCテーブル形態を決定する。

*alternate_scan (AS) — MPEG-2用フラグは、符号化及び復号化される係数の順序を決定する。

30 *vertical_size_flag (VSF) またはcontinuous_presence_multiple_point (CPM) — MPEG-1及びMPEG-2で、このフラグの映像の垂直サイズが、2800ラインを超過する場合は1にセットされ、そうでなければ0にセットされなければならない。H. 263で、このフラグは、連続的に現在のマルチポイントモードが使用されると1にセットされ、そうでなければ0にセットされる。

【0154】

【表47】

PARAM_SET1及びPARAM_SET2の定義

MPEG-1								
バイト/ビット	7	6	5	4	3	2	1	0
バイト2	予約			全_画素_順方向_ベクトル		順方向_f_コード		
バイト3	予約			全_画素_逆方向_ベクトル		逆方向_f_コード		
MPEG-2								
バイト/ビット	7	6	5	4	3	2	1	0
バイト2	順方向_垂直_f_コード				順方向_水平_f_コード			
バイト3	逆方向_垂直_f_コード				逆方向_水平_f_コード			

【0155】10. 8. 2 BP_CONTROLレジ

* いる。

スタフォーマット

【0156】BP_CONTROLレジスタフォーマット

BP_CONTROL [31:0] レジスタ (アドレス

ト

27'h7C0_0004) のためのビット仕様は、表47で示して *

【表48】

BP_CONTROLレジスタフォーマット

ビット位置	フラグ名	読出し	説明
0	BP_EN	R/W	BP 遂行イネーブル
1	SOFT_RESET	R	BP ソフトウェアリセット
2	PAUSE	R	BP 遂行中断
3	DETECT_START_CODE	R	次のスタートコード検出
4	STEP	R	BP ステップモード遂行
5	CTX_SWITCH	R	文脈切り換えリクエスト
6	CTX_MODE	R	文脈スイッチングモード
7	CTX_RELOAD	R	文脈再ロードリクエスト
8	ERR_HANDLE_MODE	R	エラー処理モード
9	—	—	予約
10	—	—	予約
16	NO_MBS[0]	R	現在スライスまたはGOBでエンコーディングされたマクロブロック数
31	NO_MBS[15]	R	

【0157】*BP_enable (BP_EN) — このフラグがARM7またはVPにより1にセットされる場合、BPはプロセッシングを遂行する。従って、すべての他のレジスタ構造は、このフラグがセットされる前に完了される。もしBPがプロセッシングを終えると、このフラグはBPによりクリアされる。

*software_reset (SOFT_RESET) — フラグがARM7またはVPによりセットされる時、BPは現在の処理を中断し、省略時の状態ですべての初期レジスタにリターンし、アイドル状態となる。ARM7はBP_ENフラグをセッティングし、BP過程をさらに始め得る。BPハードウェアリセット信号はアクチブローである。

*pause (PAUSE) — フラグがARM7またはVPにより1にセットされる時、BPは現在の処理動作を中止する。使用者はBP_ENフラグを設定することによって、中止動作が実行される。

*detect_start_code (DETECT_START_CODE) — フラグがARM7またはVPにより1にセットされる時、BPはIBUF0にあるデータの中から次のスタートコードを探す。従って、使用者がIBUF0_START及びIBUF0_ENDのための好ましいアドレスをセットしなければならない。このような命令語は、もしBPがアイドル状態であれば、適切に動作する。従って、ARM7はもしBPがアイドルでない場合、この命令を外部に送る前、ソフト

ウェアリセット命令を優先的にBPに送るべきである。

*step (STEP) - このフラグがARM7またはVPにより1にセットされると、BPは現在動作過程の一状態を遂行する。これはデバッグすることに非常に必要な特徴である。ARM7はこの段階動作がイネーブルするように中止命令を優先的に送るべきである。

*context_switching_mode (CTX_MODE) - フラグが“1”にCTX_SWITCHをセッティングし、ARM7またはVPにより“1”にセットされる時、BPは先占スイッチングモードを遂行する。もし、これはCTX_SWITCHが“1”にセッティングされることによって“0”にセットされると、BPは協力文脈スイッチングモードを遂行する。“1”にCTX_SWITCHをセッティングせず、CTX_MODEをセッティングすることはBP処理に影響を及ぼさない。文脈切換の詳細な内容はセクション10.12を参照する。

*context_reload_request (CTX_RELOAD) - フラグがARM7またはVPにより“1”にセットする時、BPは既にSDRAMに貯蔵された文脈を更にロードする。そうすると、BPはアドレスSAVE_ADR [31:0] から貯蔵された文脈を読み込む。文脈切換に対する詳細な内容はセクション10.12を参照。

error_handle_mode (ERR_HANDLE_MODE) - このフラグは、伝送された圧縮

BP_STATUSレジスタフォーマット

ビット位置	フラグ名	読出し/書き込み	説明
0	IBUF0_DONE	W	IBUF0空いている
1	IBUF1_DONE	W	IBUF1空いている。
2	IBUF0_FULL	W	IBUF0満たされている。
3	IBUF1_FULL	W	IBUF1満たされている。
4	BP_DONE	W	BP処理完了
5	CTX_SW_DONE	W	文脈切換準備
6	CTX_RELOAD_DONE	W	文脈再ロード完了
7	—	—	予約
8	BP_ERR	W	BPエラー条件
9	—	—	予約
10	IBUF0_FULL	R/W	BP入力バッファ0満たされている
11	IBUF1_FULL	R/W	BP入力バッファ1満たされている
12	OBUF0_DONE	R/W	BP出力バッファ0空いている
13	OBUF1_DONE	R/W	BP出力バッファ1空いている
14	VALID_BIT_POS[2:0]	R/W	次の動作が始まるVALID_BIT_ADR位置
15	—	—	予約
16	BP_ERR_CODE[7:0]	W	BPエラーコード

【0159】*input_buffer_0_done (IBUF0_DONE) - このフラグは入力バッファ0にあるデータが、前記のBPにより全部使用される。このフラグはBPによりセットされ、ARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*input_buffer_1_done (IBUF 50

*ビットストリームでエラーが発生した時、BPのエラー復旧過程を遂行することに利用される。入力ビットストリームが無効データである場合、BPはARM7をインタラプトさせ、このフラグの内容をチェックする。このフラグが“1”にセットされる時、BPは自動的に次のスタートコードを探す。もし、スタートコードがスライスまたはGOBであれば、BPはこの過程を更に遂行する。このフラグは“0”にセットされる時、BPは次のスタートコードを探せず、アイドル状態で動作する。BPとARM7と間のハンドシェーキングはセクション10.13に記述されている。

*number_of_macroblocks_to_be_encoded (NO_MBS [15:0]) - このレジスタは、スライスまたはGOBで符号化されたマクロブロックの数を表わす16ビットを含む。このような65535までのビット分解能を使用し、マクロブロックはスライスまたはGOBでエンコーディングされる。ここにおいて、“0”値はマクロブロック数として許容されない。

10.8.3 BP_STATUSレジスタフォーマット

BP_STATUS [31:0] (アドレス27'h 7C0_0050)は表54に示されている。

【0158】BP_STATUSレジスタフォーマット
【表49】

1_DONE) - このフラグは入力バッファ1にあるデータが、前記のBPにより全部使用されたことを表わす。このフラグはBPによりセットされ、ARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*output_buffer_0_full (OBUF0_FULL) - このフラグはBPにより出力バッ

ァ0が満たされることを表わす。前記のフラグはBPによりセットされ、そしてARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*output_buffer_1_full (OBUF1_FULL) - このフラグは、BPにより出力バッファ1が満たされることを表わす。前記のフラグはBPによりセットされ、そしてARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*BP_processing_done (BP_DONE) - このフラグは、前記のBPがスライスまたはGOBをエンコーディングするかまたはデコーディングする時、スライスまたはGOBでないスタートモードを検出したことを表わす。このフラグはBPによりセットされ、そしてARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*context_switching_done (CTX_SW_DONE) - このフラグは、BPが文脈スイッチングモードから他の作業に転換されるように準備されていることを表わす。このフラグはBPによりセットされ、そしてARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

*context_reload_done (CTX_RELOAD_DONE) - このフラグは、BPがアドレスSAVE_ADR [31:0] から貯蔵された文脈のために再ロード動作が完了されたことを表わす。このフラグはBPによりセットされ、そしてARM7またはVPによりクリアされる。このようなフラグはインタラプト状態を表わす。

【0160】*BP_error_flg (BP_ERR) - このフラグは、エラーがデータを処理する間にBPで発生されることを表わす。このフラグはBP_ERR_CODE [7:0] (BP_STATUS [31:24] がゼロでない場合にセットされる。詳細な内容はサブセクション10.9.2に記述されている。

*input_buffer_0_full (IBUF0_FULL) - このフラグは、入力バッファ0にあるデータが、ARM7またはVPにより満たされることを表わす。このフラグはARM7またはVPによりセットされ、BPによりクリアされる。

*input_buffer_1_full (IBUF1_FULL) - このフラグは、入力バッファ1にあるデータが、ARM7またはVPにより満たされることを表わす。このフラグはARM7またはVPによりセットされ、BPによりクリアされる。

output_buffer_0_done (OBUF0_DONE) - このフラグは、入力バッファ0にあ

*るデータが、ARM7またはVPにより全部使用されたことを表わす。このフラグはARM7またはVPによりセットされ、BPによりクリアされる。

*output_buffer_1_done (OBUF1_DONE) - このフラグは、入力バッファ1にあるデータが、ARM7またはVPにより全部使用されたことを表わす。このフラグはARM7またはVPによりセットされ、BPによりクリアされる。

*valid_bit_position (VALID_BIT_POS [2:0]) - 次の過程のために、VALID_BYTE_ADR [31:0] に貯蔵されている3ビット情報は、データバイトの中の有効なビット位置を表わす。ビデオエンコーディングで、BPは値を設定すべきであり、ARM7はこのビット位置から次の過程を遂行しなければならない。ビデオデコーディングで、ARM7は値を設定すべきであり、BPはこのようなビット位置からプロセッシングを遂行しなければならない。

*BP_error_code (BP_ERR_CODE [7:0]) - 8ビット情報は、BPからどのようなエラーが発生したかを表わす。ゼロ値は、何等のエラーも発生しなかったことを表わす。詳細な内容はサブセクション10.9.2で記述している。

【0161】10.8.4 デコーディングするための入力データフォーマットとエンコーディングするための出力データフォーマット

このような場合、データは実質的に圧縮されたビットストリームからなっている。このようなデータは初期コード、ヘッダーパラメータ及び、これに対応される標準によって圧縮されたデータを含まなければならない。このようなビットストリームはバイト単位別にパケットされるが、しかし、或る動作ではバイトを割当てする必要がない。このようなビットストリームは多様なスライスまたはGOBのためのデータを含む。

10.8.5 エンコーディングするための入力データフォーマットとデコーディングのための出力データフォーマット

このような場合、データは実質的にマクロブロックヘッダ情報、モーションデータ及び画素係数データからなっている。このような種類のデータ用フォーマットは、次のように定義される。

10.8.5.1 マクロブロックヘッダワード

マクロブロックヘッダは常に6バイトからなっており、テーブル33に与えられた下記のデータフォーマットを有している。

【0162】マクロブロックヘッダワードフォーマット
【表50】

マクロブロックヘッドワードフォーマット

バイト/ビット	7	6	5	4	3	2	1	0
バイト0	VMA GRNO							
バイト1	HMA MBPS							
バイト2	MT	DT M4	MB	MF	Q	P	I	
バイト3	予約			Q_SCALE				
バイト4	CBP_1		CBP_0					
バイト5	予約				CBP_1			
バイト6	予約				FID		LCI	
バイト7	予約							
バイト8	MBA_LNC (出力のみ)							
バイト9								
バイト10	PRE_DC_Y (出力のみ)							
バイト11								
バイト12	PRE_DC_Cb (出力のみ)							
バイト13								
バイト14	PRE_DC_Cr (出力のみ)							
バイト15								

【0163】ここにおいて、前記のテーブルに示したパラメータは下記に定義される。

*vertical_macroblock_address (VMA) または group_number (GRNO) - このようなバイトは、1から255までの値を有する垂直マクロブロックの位置を表わす。第1垂直位置は、0でなく1に記載されている。例外的な場合、H. 261エンコーディング時、このようなフィールドはブロックグループの位置を表わす group_number 情報を表わす。

*horizontal_macroblock_address (HMA) または macroblock_position (MBPS) - このようなフィールドは、1から255までの値を有する水平マクロブロックの位置を表わす。第1水平位置は、0でない1に記載されている。例外的な場合、H. 261エンコーディング時、このようなフィールドはGOBの中でマクロブロックの33可能性のある位置の中のいずれか1つを表わす。

*macroblock_intra (I) - もし、現在のマクロブロックが映像間符号化されると1にセットされ、そうでない場合は0にセットされる。

*macroblock_pattern (P) - もし、現在のマクロブロックが符号化されたブロックを含むと1にセットされ、そうでない場合は0にセットされる。

macroblock_quant (Q) - もし、現在のマクロブロックが新しい量子尺度パラメータを有していると1にセットされ、そうでない場合は0にセット

*される。

*macroblock_motion_forward (MF) - もし、現在のマクロブロックが順方向予測であれば1にセットされ、そうでない場合は0にセットされる。

*macroblock_motion_backward (MB) - もし、現在のマクロブロックが逆方向予測またはH. 263でB-blocksを含むと1にセットされ、そうでない場合は0にセットされる。

*dct_type (DT)、loop_filter (LF) または advanced_prediction (M4) - バイト2のビット[5]は、それぞれの動作で異なる意味を有する。MPEG-1ではこれは使用されない。MPEG-2ではdct_typeを意味する。もし、マクロブロックがフィールドDCTコーディングされた場合は、このフラグは1にセットされる。もし、フレームDCTコーディングされた場合は、0にセットされなければならない。H. 261でこのフラグはもし、ループフィルタが現在のマクロブロックで使用された場合はセットされる。そうでなければ0にセットされる。H. 263で、もし現在のマクロブロックが改良予測モードを使用した場合は1にセットされ、そうでない場合は0にセットされる。

*motion_type (MT) - 2ビットフィールドは、表56及び表57で意味する、MPEG-2で使用された frame_motion_type または field_motion_type を表わす。

【0164】

【表51】

frame_motion_typeの意味

コード	予測形態	動き_ベクトル_計算	mvフォーマット	dmv
00	予約			
01	フィールドー 基底	2	フィールド	0
10	フレームー基底	1	フレーム	0
11	デュアルー プライム	1	フィールド	1

【0165】

【表52】

field_motion_typeの意味

コード	予測形態	動き_ベクトル_計算	mvフォーマット	dmv
00	予約			
01	フィールドー 基底	1	フィールド	0
10	16×8MC	2	フィールド	0
11	デュアルー プライム	1	フィールド	1

【0166】*quantizer_scale (Q_SCALE) - DCT係数レベルの再現レベルをスケリングするために、範囲1から31まで表示されない整数である。すべてのマクロブロックヘッダは、たとえその値がその以前のマクロブロック（すなわち、macroblocks_quantはゼロである。）値と同一であるとしても、このようなパラメータに適当な値を含まなければならない。エンコーディングで、使用者はこのようなフィールドに適当な値をライトさせる責任がある。デコーディング時、BPはこのフィールドにハフマンデコードされたquantizer_scale値を書き込まなければならない。もし、現在のマクロブロックがこのフィールドにハフマンコードを含まなければ、BPはその以前のマクロブロックのスケール値を書かななければならない。

*coded_block_pattern_0 (CBP_0) - 6ビットコードは、現在のマクロブロックで符号化されたブロックを表わす。ここにおいて、
CBP_0 [5] ==> 輝度 (Y) 0ブロック
CBP_0 [4] ==> 輝度 (Y) 1ブロック
CBP_0 [3] ==> 輝度 (Y) 2ブロック

CBP_0 [2] ==> 輝度 (Y) 3ブロック
CBP_0 [1] ==> 色相ブルー (Cb) ブロック
CBP_0 [0] ==> 色相レッド (Cr) ブロック
*coded_block_pattern_1 (CBP_1) - H. 263でPBフレームのB-blocksのための追加的なcoded_blocks_patternである。ここで、
CBP_1 [5] ==> 輝度 (Y) 0ブロック
CBP_1 [4] ==> 輝度 (Y) 1ブロック
CBP_1 [3] ==> 輝度 (Y) 2ブロック
CBP_1 [2] ==> 輝度 (Y) 3ブロック
CBP_1 [1] ==> 色相ブルー (Cb) ブロック
CBP_1 [0] ==> 色相レッド (Cr) ブロック
*logical_channel_indicator (LCI) - GOB論理的なチャンネルのための2ビット情報は、ただH. 263で連続的なマルチポイントのみで使用される。

*frame_id (FID) - H. 263用のGOBフレームIDの2ビット情報

*macroblock_address_indicator (MBA_INC) - 現在のマクロブロックア

ドレスが増加する値を表わすために2バイト情報を表わす。この情報は、常に追加的な情報としてBPにより提供され、使用者は入力フォーマットに設定する必要がない。そして、入力マクロブロックヘッダワードで規定したいずれの値は、BPにより無視される。

*previous_dc_luminance (PRE_DC_Y) - 以前のマクロブロックで輝度ブロックのdc値のための2バイト情報である。もし、マクロブロックがスキップされると、リセット値が伝送される。この情報は、常に追加的な情報としてBPにより提供され、使用者は入力フォーマットに設定する必要がない。そして、入力マクロブロックヘッダワードで規定したいずれの値は、BPにより無視される。

*previous_dc_chrominance_blue (PRE_DC_Cb) - 以前のマクロブロックでブルー色彩ブロックのdc値のための2バイト情報である。もし、マクロブロックがスキップされると、リセット値が伝送される。この情報は、常に追加的な情報としてBPにより提供され、使用者は入力フォーマットを設定する必要がない。そして、入力マクロブロックヘッダワードで規定したいずれの値は、BPにより無視される。

previous_dc_chrominance_

MPEG-1及びMPEG-2のための一般的なモーションベクトルデータフォーマット

バイト/ビット	7	6	5	4	3	2	1	0
バイト0	第1順方向モーションベクトル							FS0
バイト1	水平成分							
バイト2	第1順方向モーションベクトル							
バイト3	垂直成分							
バイト4	第1逆方向モーションベクトル							FS1
バイト5	水平成分							
バイト6	第1逆方向モーションベクトル							
バイト7	垂直成分							
バイト8	第2順方向モーションベクトル							FS2
バイト9	水平成分							
バイト10	第2順方向モーションベクトル							
バイト11	垂直成分							
バイト12	第2逆方向モーションベクトル							FS3
バイト13	水平成分							
バイト14	第2逆方向モーションベクトル							
バイト15	垂直成分							

【0169】表53で、すべての要素値は、半画素精密度となる。前記のFS0、FS1、FS2及びFS3は、各モーションベクトルでフィールド選択を確認するための1ビットフラグである。もし、いずれのフィールドも選択されなければ、前記のフラグは0にセットされなければならない。その理由は、MPEG-1はフィールド選択情報を使用しないので、このようなフラグは0にセットされる。1つの例外的な場合は、デュアルプライムモーションベクトルのMPEG-2エンコーディングで発生する。このような場合、順方向モーションベクトル※

*red (PRE_DC_Cr) - 以前のマクロブロックでレッド色彩ブロックのdc値のための2バイト情報である。もし、マクロブロックがスキップされると、リセット値が伝送される。この情報は、常に追加的な情報としてBPにより提供され、使用者は入力フォーマットに設定する必要がない。そして、入力マクロブロックヘッダワードで規定したいずれの値は、BPにより無視される。

【0167】10.8.5.2 モーションデータワード

各マクロブロックヘッダはもし、マクロブロックがモーションベクトルを含むと、追加的なヘッダワードを持たなければならない。まずMPEG-1及びMPEG-2の場合を考える。このような標準は、下記の過程の中のいずれか1つが発生する時、モーションベクトルのための表59に図示された、追加的なヘッダワードフォーマットを持つようになる。

条件1) MF=1または(I=1及びCMV=1)の時、

条件2) MB=1の時、

【0168】MPEG-1及びMPEG-2のための一般的なモーションベクトルデータフォーマット

【表53】

※ルは、16バイト(実質的に8バイトが使用される)から構成され、フォーマットはテーブル37のようになる。正常的に、BPはビデオエンコーディング応用で、モーションベクトル値を差動値に変換する。しかし、テーブル37のモーションベクトル成分は、ハフマンエンコーダーですぐその入力となる差動値である。デュアルプライムモーションベクトルは、MPEG-2デコーディング応用の場合、BPにより動作される。

【0170】

【表54】

MPEG-2エンコーディング用デュアルプライムモードでモーションベクトルデータフォーマット

バイト／ビット	7	6	5	4	3	2	1	0
バイト0	順方向モーションベクトル 水平成分							
バイト1								
バイト2	順方向モーションベクトル 垂直成分							
バイト3								
バイト4	使用されない 水平成分							DMV_H
バイト5								
バイト6	使用されない 垂直成分							DMV_V
バイト7								
バイト8～15	予約							

【0171】前記のH. 261及びH. 263は、多少異なるモーションベクトルデータフォーマットを有する。大部の場合に、一バイトで或るモーションベクトル成分値を十分に表わすことができる。MF及びM4フラグの内容によって、対応されるモーション補償マクロブロックは、少なくとも2個であり、そして、多い場合1*

* 0個のモーションベクトル成分を有する。前記のモーションベクトルデータのデータフォーマットは、表60に示されている。

【0172】H. 261及びH. 263のためのモーションベクトルデータフォーマット

【表55】

H. 261及びH. 263のためのモーションベクトルデータフォーマット

バイト順	MF=1及びM4=0	MF=1及びM4=1
バイト0	順方向モーションベクトル 水平成分	第1ブロックモーションベクトル 水平成分
バイト1	順方向モーションベクトル 垂直成分	第1ブロックモーションベクトル 垂直成分
バイト2	B-ブロックモーションベクトル (MB=1であれば存在)	第2ブロックモーションベクトル 水平成分
バイト3	B-ブロックモーションベクトル (MB=1であれば存在)	第2ブロックモーションベクトル 垂直成分
バイト4	予約	第3ブロックモーションベクトル 水平成分
バイト5	予約	第3ブロックモーションベクトル 垂直成分
バイト6	予約	第4ブロックモーションベクトル 水平成分
バイト7	予約	第4ブロックモーションベクトル 垂直成分
バイト8	予約	B-ブロックモーションベクトル (MB=1であれば存在)
バイト9	予約	B-ブロックモーションベクトル (MB=1であれば存在)
バイト10-15	予約	予約

【0173】10. 8. 5. 3 ピクセル係数データワード

4個のビデオ圧縮標準は、量子化レベルのために、相異なる最大ピクセルビットの長さを有する。このような比※

※較は下記の表に示されている。

【0174】

【表56】

入出力ピクセルビット分解能

標準	量子化レベルの ための最大ビット の長さ	1つのIOBUS 読出/記入サイクル 当りバイトの#	1つのIOBUS読出 /記入サイクル当り ピクセルの#
MPEG-1	9ビット	2	2
MPEG-2	12ビット	2	2
H.261	8ビット	1	4
H.263	8ビット	1	4

【0175】従って、MPEG及びビデオ画像会議標準用のピクセルデータフォーマットは、表56のとおりその差異がある。

*【0176】ピクセル係数データフォーマット
【表57】

ピクセル係数データフォーマット

バイト順	MPEG-1及びMPEG-2	H.261及びH.2
バイト0	係数 0	係数 0
バイト1		係数 1
バイト2	係数 1	係数 2
バイト3		係数 3
バイト4	係数 2	係数 4
バイト5		係数 5
バイト6	係数 3	係数 6
バイト7		係数 7
バイト8	係数 4	係数 8
バイト9		係数 9
バイト10	係数 5	係数 10
バイト11		係数 11
バイト12	係数 6	係数 12
バイト13		係数 13
バイト14	係数 7	係数 14
バイト15		係数 15

【0177】10.9 インタラプト条件

BPはもし、このセクションで記述されたインタラプト条件に適合すると、ARM7_IRQフラグを確認しARM7をインタラプトする。前記のBPは、2個のインタラプト条件のセットすなわち、省略時及びエラー条件を有している。このような条件は、BP_STATUS[15:0]に貯蔵されている。もし、いずれか1つのビットがBPによりセットされると、ARM7_IRQ信号を活性化させる。このような条件は、BP_INT_MASK[15:0]レジスタの対応するビットをセッティングさせることによって、そのすべてをマスクすることができる。

【0178】10.9.1 省略時インタラプト条件

*Default condition 0 (BP_STATUS[0]) - 入力バッファ0の処理を終了した時、前記のBPがIBUF0_DONEフラグもセッティングさせるARM7_IRQを確認しなければならない。

*Default condition 1 (BP_STATUS[1]) - 入力バッファ1の処理を終了した

時、前記のBPがIBUF1_DONEフラグもセッティングさせるARM7_IRQを確認しなければならない。

*Default condition 2 (BP_STATUS[2]) - 入力バッファ0の処理を終了した時、前記のBPがOBUF0_DONEフラグもセッティングさせるARM7_IRQを確認しなければならない。

*Default condition 3 (BP_STATUS[3]) - 入力バッファ1の処理を終了した時、前記のBPがOBUF1_DONEフラグをセッティングさせるARM7_IRQを確認しなければならない。

*Default condition 4 (BP_STATUS[4]) - ビデオエンコーディングの場合、ARM7により設計されたスライスまたはGOBを終える時、またはビデオデコーディングの場合、スライスまたはGOBでなく初期コードに到着した時、また前記の

BPはBP_DONEフラグをセッティングするARM7_IRQを確認しなければならない。

*Default condition 5 (BP_STATUS [5]) - 先占文脈スイッチングモードで文脈貯蔵動作を終了した時、または協力文脈スイッチングモードで、現在のスライスまたはGOBを終了した時、前記のBPはCTX_SW_DONEフラグをセッティングするARM7_IRQを確認しなければならない。

*Default condition 6 (BP_STATUS [6]) - 文脈再ロードが終わった時、前記のBPは、CTX_RELOAD_DONEフラグをセッティングするARM7_IRQを確認しなければならない。

*Default condition 7 (BP_STATUS [7]) - 現在、BP_STATUS [7] が維持される。従って、このようなビットは“0”にセットされるべきである。正常的に、このような省略時インタラプト条件は、BP_INT_MASK [7:0] を使用しマスクされるように勧告はしない。しかし、或る動作では、使用者がDefault condition 1をマスクすることを願う場合もある。

【0179】10.9.2 エラーインタラプト条件
もしエラーがBPで発生すると、前記のBPはARM7インタラプトが要求されるように、BP_ERRフラグをセットする。同時に、前記のBPは、前記のBP_STATUSレジスタのBP_ERR_CODEフィールドでゼロでない値の中から適当なデータをセットする。このような8ビットBP_ERR_CODEは下記のような意味を有している。

*BP_CODE=8'b0000_0000: 何等のエラーも発生しない

*BP_ERR_CODE=8'b0000_0001: BP_MODEレジスタに不適当にセッティングされる

*BP_ERR_CODE=8'b0000_0010: 不適当に水平マクロブロック位置がセットされる

*BP_ERR_CODE=8'b0000_0011: 不適当に垂直マクロブロック位置がセットされる

*BP_ERR_CODE=8'b0000_0100: マクロブロックアドレス増加に対する不適当したVLC

*BP_ERR_CODE=8'b0000_0101: マクロブロックタイプに対する不適当したVLC

*BP_ERR_CODE=8'b0000_0110: マクロブロックモーションコードに対する不適当したVLC

*BP_ERR_CODE=8'b0000_0111: 不適当した取消しモーションベクトルマーカビット

*BP_ERR_CODE=8'b0000_1000: 符号化されたブロックパターンに対する不適当したVLC

*BP_ERR_CODE=8'b0000_1001: ブロックDCT_dcサイズに対する不適当したVLC

*BP_ERR_CODE=8'b0000_1010: 不適当した

DCT_dc値

*BP_ERR_CODE=8'b0000_1011: ブロックDCT_ac係数に対する不適当したVLC

*BP_ERR_CODE=8'b0000_1100: 1つのマクロブロックでブロックの#が64を超過する

*BP_ERR_CODE=8'b0000_1101: 不適当したf_CODE値 (例えば、値が0)

*BP_ERR_CODE=8'b0000_1110: ブロックDCT_ac係数に対する不適当したVLC

*BP_ERR_CODE=8'b0000_1111: 不適当したIBUF及びOBUFアドレス設定

*BP_ERR_CODE=8'b0001_0000: BP入出力バッファ用スタートアドレスの最下位4ビットはゼロでない

他のBP_ERR_CODE値は貯蔵される。

【0180】10.10 詳細な機能性要求

10.10.1 IOBUSインターフェース

BPとCCUとの間のすべてのデータ移動は、IOBUSを通して遂行される。前記のIOBUSは、多重化されたアドレス及びデータを含む32ビット@40MHz同期バスである。少なくとも7サイクルは、前記のIOBUSを通して16バイトデータを伝達するように要求されるので、前記IOBUSの最大伝達速度は、91.4Mbytes/sec(731.4Mbits/sec)である。前記のBPは、すべてのIOBUS読出及び書込伝送のためにマスタまたはスレーブとなり得る。前記のBPがマスタとして動作する時、IOBUSアービタにリクエスト信号を送らなければならない。もし、前記のIOBUSがなければ、前記のアービタは前記のBPに与え、そしてデバイスセレクト信号を送る。前記のIOBUSを通したデータの競争は、下記の3個のカテゴリーの中のいずれか1つである。すなわち、2または4このピクセル要素を含んでいる32ビットピクセルデータと、32ビット圧縮ビットストリームワードと、エンコーディング及びデコーディング動作のための構文/制御パラメータの中の1つである。IOBUSインターフェースに関するタイミング図のような情報に追加し、使用者は前記のMSP IOBUS仕様を検討するよう勧告される。

10.10.2 ブロック層プロセッシング

10.10.2.1 ジグザグスキャン規定

前記のBPは、MPEGビデオ標準で提示された2個のジグザグスキャン変換マトリクスを支援する。VP及びBPの間で伝送されるこのような8×8ブロックデータは、すべて64個の成分を含んでいる。

10.10.2.2 RLCコード

RLCデコーディングのために、前記のBPはDCT_ac係数のハフマンデコード結果によって、ゼロ及びレベルデータを発生する。もし、データが1つの8×8ブロックに生成される前、64個のピクセルend_of_block信号が検出されると、前記のRLCデコー

ダーは、残りのゼロデータを作るようになる。RLCエンコーディングのために、前記のBPは隣接したゼロデータをカウントし、次のNON-ゼロデータと結合することによって、ラン-長さ及びレベルコードを発生させる。もし、残っているデータのすべてがゼロに等しい場合、残っているデータのためにRLCを発生するよりは、end_of_blockを発生させるとよい。RLCコードのための動作サイクルは、このように発生されたゼロの数ほど進行される。

【0181】 10. 10. 2. 3 ハフマンコード 10 【0182】

前記のBPハフマンコードは、MPEG-1、MPEG* 【表58】

ハフマンデコーダールックアップテーブルのために要求されるロムサイズ

NO	標準	テーブル説明	ROMサイズ (ビット)
1	MPEG-1&2、 H.261	dct_coeff_next、TCOEFF	144 × 12
2	H.263	TCOEFF	102 × 12
3	MPEG-1&2、 H.261、H.263	motion_vector_code macroblock_address_increment	17 × 12
4	MPEG-1&2、 H.261	coded_block_pattern	32 × 12
5	MPEG-1&2、 H.261、H.263	macroblock_type及びMCBPC	18 × 12
6	MPEG-1&2	Y及びCbCr用のdct_dc_size	12 × 12
7	H.263	イントラー及び内部フレームのためのCBPY	8 × 12
合計			333 × 12

【0183】 前記のエンコーダテーブルは、前記のデ
コーダテーブルより更に大きいロムサイズを要求する
内容が、表に要約されている。 ※

※ 【0184】

【表59】

ハフマンエンコーダールックアップテーブルのために要求されるロムサイズ

NO	標準	テーブル説明	ROMサイズ (ビット)
1	MPEG-1&2、 H.261	dct_coeff_next、TCOEFF	156 × 12
2	H.263	TCOEFF	98 × 12
3	MPEG-1&2、 H.261、H.263	motion_vector_code、 macroblock_address_increment	35 × 12
4	MPEG-1&2、 H.261	coded_block_pattern	64 × 12
5	MPEG-1&2、 H.261	macroblock_type	22 × 12
6	H.263	MCBPC	28 × 12
7	MPEG-1&2	Y及びCbCr用のdct_dc_size	16 × 12
8	H.263	イントラ及び内部フレームの ためのCBPY	16 × 12
合計			435 × 12

【0185】表から、ハフマンエンコーダ及びデコーダのために全体的に要求されるロムのサイズは、768×3012ビットである。前記のテーブルは、スタッフコード、escape_code、DCT係数のサインビット及び、ステートマシンにより操作されるend_of*
*_blockコードを含んでいない。前記それぞれのハフマンコードのための動作サイクルは、表に記述されている。
【0186】
【表60】

ハフマンコード用処理サイクル

ハフマン コード形態	MPEG-1&2に対する 処理サイクル	H. 261&H. 263に 対する処理サイクル	処理クロック
dct差動 dc係数	4-5	5-6	40MHz
補数を有する dct ac係数	1	2	
エスケープコ ードプラス固 定長さコード	2	3	
end_of_block コード	1	2	
他のハフマン コード	1	1	

【0187】最後に、J P E Gデコードテーブルは、前記のような接近過程を用いると、具現され得ないことを表わす。しかし、d c _ c o e f f _ n e x t _ 0テーブルでは、J P E Gエンコーディング応用が使用され得る。

10.10.2.4 差動d c値

イントラブロックの場合、前記のB Pはまた、8×8ブロックデータの第1番目要素の差動d c係数を計算し、そして既に伝送された差動d c係数でd c値を再現する。 30

10.10.2.5 非コード化されたブロック

前記のB Pは、コード化されないブロックを支援しない。前記のV P及びA R M 7は、非符号化されたブロックを遂行する。前記のV P及びA R M 7がこのようなブロックを処理するように、前記のB Pはマクロブロックヘッダのワードに表れているc o d e d _ b l o c k _ p a t t e r nで、非コード化されたブロックを表わす。

10.10.2.6 ブロック伝送順序

エンコーディング及びデコーディングのために伝送された1つのマクロブロックで、ブロックの順序は下記のとおりである：輝度（Y）ブロック0、1、2及び3、色相ブルー（Cb）及び色相レッド（Cr）ブロックである。

10.10.3 マクロブロック層処理

10.10.3.1 差動モーションベクトル

前記のB Pは、前記のモーション推定結果から差動モーションベクトルを計算し、次の場合を除いては伝送された差動モーションベクトルを有してモーションベクトルを再現する。

*第1場合は、M P E G - 2ビデオエンコーディングの場合のデュアルプライムモードである。この場合で、前記のB Pに伝送されたモーションベクトルは、ベクトル‘[0] [0] [1:0]’であり、ベクトル‘[r] [0] [1:0]’ではない（M P E G - 2ビデオ標準の7.6.3.6節参照）。

*第2の場合は、H. 263の改良予測モードである。この場合、4個のモーションベクトル及び、このような値が差動値として前記のB Pから／に伝送されなければならない。

10.10.3.2 スキップされたマクロブロック

前記のB Pは、スキップされたマクロブロックを支援しない。前記のV P及びA R M 7は、このようにスキップされたマクロブロックを処理する。前記のようにスキップされたマクロブロックを処理するための前記のV P及びA R M 7で、前記のB Pは、前記マクロブロックのヘッダワードに水平及び垂直マクロブロックアドレスを書き込む。

40 10.10.3.3 マクロブロックスタップコード

M P E G - 1で、一サイクルでマクロブロックスタップコードがもし生ずると、前記のB Pはこれを捨てなければならない。しかし、M P E G - 1エンコーディングで、B Pは使用者がマクロブロック層ヘッダ内のマクロブロックスタップコードを含まないようにする。一般的に、このようなスターピングコードは、出力ビデオレートバッファを制御することに使用される。従って、マクロブロックスタップコードを挿入する代わりに、スタートコードの間にゼロスタップビットを挿入するように勧告する。M P E G - 1及びM P E G - 2の応用のため 50

に、ビットストリーム出力はスライス層までバイトーアラインメントされなければならない。たとえビットストリーム出力が画像層までバイトーアラインメントされてもH. 263応用のためには、GOB層までバイトーアラインメントさせる。しかし、H. 261エンコーダーの出力はバイトーアラインメントされない。従って、ARM7でルーチンを形成するビットストリームは、このような差異を考慮してプログラムされる。IOBUSを通した最後のデータ伝送のためのデータ量が、エンコーディングの場合16ビット以下の場合、BPはスライスの 10
 エンドにゼロを満たす動作(zero-fill)を自動的に遂行する。

【0188】10.10.4.2 エキストラスライス情報

デコーディングにおいて、BPはMPEG-1またはMPEG-2ビットストリームのスライスヘッダに含まれる、任意のエキストラスライス情報を捨てる。エンコーディングにおいて、BPは使用者により要請された任意のエキストラスライス情報を挿入させない。もし、使用者がまだMPEG-1またはMPEG-2ビットストリー 20
 ムにこの情報を含ませようとする場合は、BPにより予めエンコーディングされたビットストリームにこの情報を挿入すればよい。

10.10.4.3 イントラスライスMPEG-2スライス層ビットストリームにおいて、intra_sliceというパ

ラメータは、現在スライスがイントラマクロブロックのみで構成されたことを表わすことに使用される。この情報は、デコーディング過程では使用されず、高速前進または高速後進機能を遂行する場合、DSM応用を助け 30
 るためのものである。従って、BPはデコーディング応用の場合この情報を捨て、エンコーディング応用の場合、スライス層ヘッダにあるintra_sliceに0を挿入する。 10.10.4.4 スライスまたはGOBスタートコード

MPEG-1、MPEG-2、H. 261において、すべてのピクチャーは少なくとも1つのスライスまたはGOBスタートコードを有する。しかし、H. 263ピクチャーは、GOBスタートコードとヘッダ情報を有しない。特に、任意のH. 263ピクチャーで第1番目のGOBは、スタートコードとヘッダ情報を有しない。従って、入力されるビットストリームがH. 263のためのものである場合、BP状態マシンは、マクロブロック層を直ちに処理しなければならない。それだけでなく、ビットストリームがデコーディングされる間、GOBスタートコードが発見されると、BPはスタートコードをデ 40
 コーディングし、ARM7をインタラプトせず処理過程を続いて遂行する。

【0189】10.11 入力/出力ダブルバッファインターフェース

10.11.1 一般的な説明

入力及び出力バッファは、ダブルバッファにより具現される。従って図64と図65の図示のとおり、IBUF0、IBUF1、OBUF0、OBUF1の4個のメモリバッファを使用するようになる。図64よ図65のように、それぞれのバッファはスタート及び終了アドレスと充満及び完了フラグを有する。各バッファサイズを決定するために、使用者は各バッファに対するスタート及び終了アドレスに適切な値を記入しなければならない。一応、バッファ用ソースプロセッサは、バッファに対する記入を完了すると、充満フラグを設定し、他のバンクに対する記入を開始する。バンク用シンクプロセッサはアクセスされるバンクが全部満たされていることが分かると、データを読み出す。バンクが空いていると、シンクは完了フラグを設定し、他のバンクの充満フラグをチェックする。4個のスタートアドレスは、小節10.7.2の説明のとおり、BPによりアップデートされる。スタートアドレスのためのそれぞれのレジスタは、BPが入力または出力バッファをアクセスする都度に、BPによりアクセスされる最後のバイトアドレスを 50
 貯蔵する。従って、ARM7はIBUF0_DONE、IBUF1_DONE、OBUF0_FULL、OBUF1_FULLのフラグの中のいずれか1つが設定されると、該当スタートアドレスを設定する。また、スタートアドレスの最後の4ビットは常にARM7によりゼロに設定される。その理由はFBUS、CCUとIOBUSとの間の内部データ割り当て構造のためである。また、各バッファサイズの全体バイト数が16の倍数となるように、それぞれの最後アドレスを設定しなければならない。それだけでなく、最小バッファサイズは、MPEG-1とMPEG-2に対して64バイト、H. 261とH. 263に対して128バイトであることが勧告される。これはARM7に対するBPの頻繁なインタラプトによる、性能の劣化を防止するためのことである。

【0190】10.11.2 非正常的なバッファステータスの処理

2個の出力バッファが満たされると、BPは処理を中止させ、入力ダブルバッファステータスに関係なく、アイドル状態に落ちる。OBUF0_DONEまたはOBUF1_DONEフラグが設定されると、BPは自動的にこのアイドル状態から抜け出る。2個の入力バッファが空くと、BPは処理を中止する必要がなく、内部に残っているデータの処理が完了されるまで処理し続ける。しかし、2個の入力バッファが空くと、BPはすぐARM7をインタラプトする。残っているデータ処理の終了以降、入力バッファがまだ空いていると、BPはアイドル状態に落ちる。IBUF0_FULLまたはIBUF1_FULLフラグが設定されると、更にBPは自動的にこの状態から抜け出る。この小節で記述するアイドル状態は、この仕様書で記述した他のアイドル状態とは異な 50

る。その理由は、他のアイドル状態から抜け出するためには、通常ARM7の制御コマンドを必要とするからである。

【0191】10.11.3 I/Oバッファの物理的な具現：例

大抵、BP入力及び出力バッファの位置とサイズを決定することは、使用者の分である。使用者はVPデータキャッシュ、ARM7データキャッシュまたはSDRAMのスクラッチパッド領域でバッファを具現する。BP入力及び出力ダブルバッファの具現が多少制限的であって 10
も、前記のバッファを具現するための効率的な方法がある。ここにおいて、ビデオデコーディング応用において、レートバッファの具現に対する特殊な例を挙げる。この場合、使用者はBP入力バッファを循環バッファで具現しようとする。ここで、SDRAMを使用し、完全なレートバッファは図66の図示のとおり、4個のブロックに分割されるものと仮定する。初期に、使用者はRate_Buffer_Block_0とRate_Buffer_Block_1とをそれぞれIBUF0とIBUF1に設定することができる。これは下記のように設定することによって可能になる。 20

```
IBUF0_START=Rate_Buffer_Address_0;
```

```
IBUF0_END=Rate_Buffer_Address_1;
```

```
IBUF1_START=Rate_Buffer_Address_2;
```

```
IBUF1_END=Rate_Buffer_Address_3.
```

IBUF0にあるデータ（すなわち、Rate_Buffer_Block_0にあるデータ）のすべてがBPにより使用されると、BPはARM7をインタラプトする。そうすると、ARM7は下記のように設定することによって、Rate_Buffer_Block_2をIBUF0に設定する。 30

```
IBUF0_START=Rate_Buffer_Address_4;
```

```
IBUF0_END = Rate_Buffer_Address_5.
```

IBUF1にあるデータのすべてがBPにより使用されると、BPはARM7をインタラプトする。そうすると、ARM7は下記のように設定することによって、Rate_Buffer_Block_3をIBUF1に設定する。 40

```
IBUF1_START=Rate_Buffer_Address_6;
```

```
IBUF1_END=Rate_Buffer_Address_7.
```

Rate_Buffer_Block_2にあるデータのすべてがBPにより使用されると、ARM7は第1番 50

目の段階でのようにアドレスを設定することによって、Rate_Buffer_Block_0をIBUF0に更に設定する。従って、循環バッファは単にこのような完全な過程を繰り返すことによって具現され得る。この例は、BPダブルバッファの使用が、使用者の意図によって非常に柔軟であることを表している。

【0192】10.12 文脈スイッチング

1つ以上の応用がMSPを駆動させる場合、ARM7動作システムは、BPが現在作業を終結させ、他の作業に切り換えられるように命令する。この過程は、通常に“文脈スイッチング”という。BPは下記の2種類の文脈スイッチングモードを支援する。

10.12.1 先占(preemptive)文脈スイッチング

先占文脈スイッチングは、BPが現在8×8画素ブロック処理を遂行してから正常的な処理が終了したことを意味する。ARM7はBP_CONTROL[6:5]レジスタにあるCTX_SWITCHとCTX_MODEフラグを“11”に設定することによって、先占文脈スイッチングモードを命令する。現在ブロック処理が完了されると、BPは後の処理のために、内部文脈を外部SDRAMに送る。BPは文脈貯蔵を完了すると、BP_STATUS[5]に位置したCTX_SW_DONEフラグを設定することによって、ARM7をインタラプトする。そうすると、ARM7はBPの入出力バッファのすべての内容を貯蔵し、他の作業のためにBPを初期化させる。このモードは、BPがARM7の文脈スイッチングリクエストをできるだけ速く応答するようにする。最悪の場合、BPは現在ブロック処理を完了するために、約150サイクル(=3.75μsec)を必要とする。しかし、正常的な場合、ブロック処理を完了するために数十のサイクルを必要とすると見なすことが好ましい。

【0193】10.12.2 協調型(cooperative)文脈スイッチング

協調型文脈スイッチングによると、BPで文脈貯蔵過程を除去することができる。これはすべてのスライスまたはGOB層処理時に、BP内部の状態のすべてを初期化させなければならないという事実起因したものである。このモードで、BPは現在スライスまたはGOBを続いて正常的に処理してから処理を完結させる。ARM7はBP_CONTROL[6:5]レジスタにあるCTX_SWITCHとCTX_MODEフラグを“10”に設定することによって、協調型文脈スイッチングモードを命令する。現在スライスまたはGOB処理が完了されると、BPはBP_STATUS[5]に位置したCTX_SW_DONEフラグを設定することによってARM7をインタラプトする。そうすると、ARM7はBPの入出力バッファのすべての内容を貯蔵し、他の作業のためにBPを初期化させる。

【0194】10.12.3 文脈再ロード

その以前の作業を切換するために、BPはアドレスSA

VE_ADR [31:0] からSDRAMに貯蔵される文脈を再ロードする。この文脈再ロードを要請するために、BPがアイドル状態にある必要がある。このリクエストのための可能な状況は、BP_DONEが設定された場合、CTX_DONEまたはARM7がソフトウェアでBPをリセットさせた場合である。それで、ARM7がBP_CONTROL [7] にあるCTX_RELOADフラグを設定すると、BPはアイドル状態から抜け出て、貯蔵された文脈を読み出し始める。BPは文脈再ロード動作を完了してから、CTX_RELOAD_DONEフラグを設定してARM7をインタラプトする。そうすると、ARM7はBPの内部レジスタを初期化させ、以前の作業処理のためにBPをイネーブルさせる。

10. 13 作業ハンドシェーキング

この節は、BPが処理を終えた場合、作業ハンドシェークのための細部的な過程を取扱う。ここにおいて、“最後のデータのためのポインターのアップデート”とは、BPがVALID_BYTE_ADR [31:0] とVALID_BIT_POS [2:0] にそれぞれ適切な

値を記入したことを意味する。
【0195】10. 13. 1 エンコーディングの場合
正常状態で、エンコーディングのための入力データは、VPから供給される。入力ダブルバッファの中の1つがVPにより満たされると、BPはIOBUSを通してデータの脱出を開始する。処理が終了される時点（すなわち、処理されたマクロブロック数が、ARM7により指定されたマクロブロック数と同一な場合）で、BPはBP_DONEフラグを設定してARM7をインタラプトし、アイドル状態に落ちる。有効データのためのポインターは、スライスまたはGOBに対する“圧縮されたビットストリームの最後”を表わす。また、VALID_BYTE_ADR [31:0] は、出力ダブルバッファの中のいずれか1つの位置を表わす。ARM7はこの圧縮されたビットストリームと、上位層ヘッダを結合させて最終のビットストリームを形成し、処理過程を繰り返す。出力ダブルバッファにあるデータを完全に焼尽させる前、ARM7がBPを再開しようとする場合は、少なくとも1つの出力ダブルバッファを焼尽させ、BPが再開された時、ポインタはBPによりアップデートされるので、最後のデータのためのポインタはそのまま置いておくことによって可能である。

【0196】10. 13. 2 デコーディングの場合
まず、ARM7はスライスまたはGOBスタートコード（存在時）を探索する。スタートコードが発見されると、ARM7はBPを初期化しイネーブルさせる。BPでハフマンデコーディング、RLCデコーディング及び逆ジグザグスキャン変換を遂行してから、データはVP処理のために出力バッファに伝送される。BPはこの処理ルーチンを非スライスまたは非GOBスタートコード

が発見されるまで続く。これらが検出されると、BPは“非スライスまたは非GOBスタートコードのエンド”に使用される最後のデータのためにポインタを設定し、ARM7をインタラプトする。次に、ARM7はスタートコードをデコーディングして、次のスライスまたはGOBコードが発見されるまで、ヘッダ解釈(parsing)を遂行する。

10. 13. 3 圧縮されたビットストリームで発見されたエラー

実際のデータが、電話線及び公衆スイッチ網を通して伝送される画像電話応用において、いくつかの無効データが入力されるビットストリームに含まれる可能性が非常に高い。この場合、BPはARM7をインタラプトし、ERR_HANDLE_MODEフラグをチェックしなければならない。もしBPが特定の応用のためにイネーブルされる前、使用者がエラー処理モードを決定する場合は、安全である。ERR_HANDLE_MODEフラグが“1”に設定されると、BPは自動的に次のスタートコードを探し出す。スタートコードがスライスまたはGOBのものである場合、BPは正常的な処理を続ける。このモードが非常に効率的であるが、その理由はBPがARM7より更に迅速にスタートコードを探し出すことができ、BPが次のスタートコードを探す間、ARM7が他の処理ルーチンを遂行し得るからである。しかし、スライスまたはGOB層とは異なるスタートコードが発見されると、BPはBP_DONEフラグを設定してARM7を更にインタラプトし、アイドル状態に落ちる。このような場合、最後のデータのために使用されたポインタは、次のスタートコードのエンズを指示しなければならない。ERR_HANDLE_MODEフラグが“0”に設定されると、BPは次のスタートコードを探せず、アイドル状態に落ちる。このような場合、最後のデータのために使用されるポインタは、エラーは発見された位置を指示しなければならない。このモードは使用者がARM7命令語を利用して、汚染されたビットストリームをデバギングする場合に有用である。

【0197】

【実施例2】

付 録 B

MPCビットストリーム処理器

ビットストリーム処理器（BP）は、ビデオデータエンコーディング及びデコーディング応用に重要なMSP処理コアの中の1つである。BPはMPEGスライス層エンコーディング及びデコーディング、そしてH. 261/H. 263グループブロック（GOB）層エンコーディング及びデコーディングを処理する。デコーディング応用において、BPは各マクロブロックに含まれた全体情報を、ベクトル処理器及びARM7コアに提供する。ビットストリーム処理器ハードウェアは、4個の機能ブロックに分けられる。

* I/O制御及びデコーディングユニットを含むIOBUSポートインターフェース

* BP制御ステートマシン

* BPレジスタマルチプレクサー、レジスタ、算術論理ユニット(ALU)及びマルチプレクサー、FIFO制御ユニットを含むコーデックコア

* VLC FIFOユニット

* コーデックアドレス発生器と共にルックアップを含むVLCコーデック

VLC LUT ROM (340、図3参照) についての説明は下記のとおりである。

1.0 方法論

ルックアップユニットは、ハフマンエンコーディング及びデコーディングの核心である。このユニットは、MPEG-1、MPEG-2、H.261及びH.263仕様に含まれたVLCテーブルを支援し、三星MSPにより支援される。このテーブルの大部分は、12ビット幅を有するROMで具現される。しかし、ルックアップ処理があまり単純で、ROMテーブルのサイズに当たらない場合、特殊なエンコーディング及びデコーディングが適用される。このようなレイヤーの4つの仕様のすべては、多い可変長さコードを17ビットまで含む。エンコーディング値またデコーディング以外に、コードサイズ及び有効コード指示者が、エンコーディング及びデコーディングのために提供され、正確に処理されるようにする。VLCテーブルをエンコーディングまたはデコーディングするために従来の方法を使用する場合、ROMテーブル及びアドレス発生器が更に大きくなる。

【0198】 1.1 具現方法は下記のとおりである：

* もしアドレス発生器の設計が難しくなければ、ROMテーブルをできるだけ多く共有する。

* エンコーディングまたはデコーディングに基づいて、VLCテーブルを再配列する。

*

ハフマンデコーダールックアップテーブルのために要求されるROMサイズ

NO	標準	テーブル説明	ROMサイズ (ビット)
1	MPEG-1&2、 H.261	dct_coeff_next、TCOEFF	144 × 12
2	H.263	TCOEFF	102 × 12
3	MPEG-1&2 H.261、H.263	motion_vector_code macroblock_address_increment	17 × 12
4	MPEG-1&2、 H.261	coded_block_pattern	32 × 12
5	MPEG-1&2 H.261、H.263	macroblock_type及びMCBPC	18 × 12
6	MPEG-1&2	Y及びCbCr用のdct_dc_size	12 × 12
7	H.263	イントラ及び内部フレームのためのCBPY	8 × 12
合計			333 × 12

【0200】 VLCエンコーディングROMテーブルマ ュップ ※【表62】

※

* *ハフマンコードに基づいて‘0’カウントと‘1’カウントをまずデコーディングする。

* 符号または偶数/奇数のような1ビットフラグを使用しテーブルサイズを減らす。

* 可能であれば、1つのROM位置を‘上位’と‘下位’に分離する。

* ROMテーブルアドレスを発生させるために、VLCの最下位ビット(LSB)を使用しアドレス発生器を簡素化する。

この方法は非常に効率的である。最終のROMテーブルサイズは、768×12ビットで、問題を伴うにはずつと小さい。ルックアップは、ROMテーブルアドレス発生器と、ROMテーブルルックアップ処理により遂行される。アドレス発生器はテーブル種類、モード及びVLC値のような入力信号をデコーディングし、ROMテーブルのアドレスを発生させる。以降、ROMテーブル値及び他の情報からエンコーディングまたはデコーディングデータが得られる。デコーディングテーブルは2つのフォーマットを有するが、一つはVLCコード当りのROM位置を有するDCT係数に適用され、もう一つはそれぞれのROM位置が上位6ビットと下位6ビットに分割されている他のテーブルに適用される。従って、各位置は2つのVLCコードを有する。エンコーディングテーブルは、2つのフォーマットを有するが、一つはH.263のTCOEFFに関するもので、もう一つは他のテーブルに対するものである。各ROM位置はエンコーディング応用のために一つのハフマンコードを含む。ROMテーブルのサイズは768×12ビットである。テーブルは下記のように示すことができる。

【0199】 VLCデコーディングROMテーブルマップ

【表61】

ハフマンエンコーダールックアップテーブルのために要求されるロムサイズ

NO	標準	テーブル説明	ROMサイズ (ビット)
1	MPEG-1&2、 H. 261	dct_coeff_next、TCOEFF	156 × 12
2	H. 263	TCOEFF	98 × 12
3	MPEG-1&2、 H. 261、H. 263	motion_vector_code、 macroblock_address_increment	35 × 12
4	MPEG-1&2、 H. 261	coded_block_pattern	64 × 12
5	MPEG-1&2、 H. 261	macroblock_type	22 × 12
6	H. 263	MCBPC	28 × 12
7	MPEG-1&2	Y及びCbCr用のdct_dc_size	16 × 12
8	H. 263	イントラ及び内部フレームの ためのCBPY	16 × 12
合計			435 × 12

【0201】1. 2 デコーディング

デコーディングに対するすべてのテーブルは、'0' または '1' カウントに基づいて再配列される。VLCコードのMSBが'0'の場合は'0'カウントが印加され、'0'でない場合は'1'カウントが使用される。例えば、コード'00001xxx'の場合4個の'0'を有し、コード'1110xxx'の場合3個の'1'カウントを有する。デコーディング過程はまず'0'/'1'カウントをデコーディングし、VLCコードの'0'/'1'カウントをROMテーブルアドレス発生器へ出力する。以降、アドレス発生器は残りのコードをデコーディングしてアドレスを発生させる。アドレスは2個部分けられるが、一つはオフセットで、もう一つはいわゆるマスクされたアドレスであって、VLCテーブルから得られる。アドレスは2個部分に対する論理和(OR)から得られる。アドレス発生器により提供される他の情報は、下記の通り示すことができる。

*VLC code size

*Special Flag: 2ビットフラグは、H. 261で'ESCAPE'、'END OF BLOCK'、'STUFFING'、または'START CODE'に対するデコーディングステートマシンを表わす。

*High data extract enable: 有効データは上位6ビットである。

*Sign/even enable: このフラグは、デコーディングがVLCのLSBを、テーブルに基づいた符号または偶数ビットで抽出するべきであることを指示する。

*Valid VLC

*Mask shift bits 及びmask: この両信号は、マスクされたアドレスを発生させるために印加される。

ROMテーブルに対して、MPEG-2のテーブル14、15とH. 263のテーブル12を除外しては、それぞれの位置に上位と下位ビットフォーマットから形成

されたデータが貯蔵される。

【0202】1. 2. 1 テーブル12/MPEG-2
このテーブルはテーブル2-B. 5c/MPEG-1及びテーブル5/H. 261と同一である。

ROMテーブルフォーマット: ビット10~6: ラン; ビット5~0: レベル

1. 2. 2 テーブル15/MPEG-2

テーブル14/MPEG-2と同一なラン、レベル及びVLCコードを有するので、このテーブルの大部分は、テーブル14/MPEG-2と共有する。

ROMテーブルフォーマット: ビット10~6: ラン; ビット5~0: レベル

1. 2. 3 テーブル12/H. 263

このテーブルは、MPEG-2のテーブル14、15と比較する時、1個以上の出力値'LAST'を有する。

ROMテーブルフォーマット: ビット11: LAST; ビット10~4: ラン; ビット3~0: レベル

1. 2. 4 モーションコード/マクロブロック増加分

この節は、テーブル1/MPEG-2、テーブル10/MPEG-2、テーブル2-B. 1/MPEG-1、テーブル2-B. 4/MPEG-1、テーブル1/H. 261及びテーブル3/H. 261及びテーブル10/h. 263を取扱う。

【0203】モーションコードに対して、VLC=1の場合を除いてはLSBが符号ビットである。マクロブロック増加分に対して、LSBはVLC=1の場合を除いては、偶数値フラグである。従って、テーブルの半分のみをデコーディングする。タイル(tile)符号/偶数ビットを無視する場合、テーブル10/H. 263の上位部分を除いては、2種類のテーブルは、同一なVLC値とデコーディング値を有する。デコーディングされた値は、6ビットまで発生し、これは2個のテーブル値を1つの位置に置くことができることを意味する。例えば、テーブル10/H. 263の下位部分のデコーディング値が他のものと異なっても、タイル2進値は固定小数点のために同一である。すなわち、このすべてのテーブルを

169

取扱うために、固定小数点として16 半分の位置を使用する。ROMアドレスを発生させるために、1つの簡単なFSMを使用する。応用において、モーションコードがデコーディングされる場合、ROMテーブルは絶対値を提供する。一方、アドレス発生器が符号ビットをイ*

```
if (sign_enable==1)
    increment_value=sign*ROM_table_value;
else
```

もし、マクロブロックアドレス増加テーブルがデコーディングされると、その結果はROMテーブル値と偶数フラグから得られる。例えば、ROMテーブルは‘5’の値を提供する。偶数フラグが‘ハイ’であれば‘10’※

```
if (even_enable==1)
    increment_value=(ROM_table_value
e<<1)
| (~even_bit);
```

```
else
```

```
    increment_value=ROM_table_value
; ROMテーブルフォーマット：ビット11～6：上位データ；ビット5～
0：下位データ
```

1. 2. 5 マクロブロックパターン

この節はテーブル9/MPEG-2、テーブル2-B、3/MPEG-1、テーブル4/H. 261 (CBP)を取扱う。デコーディングされた値は6ビットまで発生するが、これは1個の位置に2個のデータを置くことができることを意味する。すなわち、このテーブルのすべてを取扱うためには、32個の位置が使用される。ROMテーブルフォーマット：ビット11～6：上位データ；ビット5～0：下位データ

【0204】1. 2. 6 マクロブロックタイプ

この節は、テーブル2、3、4/MPEG-2、テーブル2-B、2/MPEG-1、テーブル2/H. 261 (MTYPE) 及びテーブル3、4/H. 263 (MCBPC)を取扱う。デコーディングされた値は5ビットまで発生する。ここでも上位/下位データ概念を使用する。ROMアドレスを発生させるために、1個の簡単なFSMが使用される。

★40

MPEGのマクロブロックタイプフォーマット

ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
予約	Q	FMV	BMV	P	1

【0206】

【表64】

170

* ネーブルさせると、デコーダーはLSBを抽出するが、‘1の場合’は負(-)、‘0’は正(+)を意味する。このアルゴリズムは下記のように示すことができる。

※の結果が得られ、偶数フラグが‘ロー’であれば‘11’の値が得られる。このアルゴリズムは下記のように示すことができる。

★ROMテーブルフォーマット：ビット11～6：上位データ；ビット5～0：下位データ

たとえ、いくつかのビットが各仕様によって相異なる意味を有するが、マクロブロックタイプのフォーマットは、MPEGに基づいて各仕様に対して普遍的に定義されている。H. 263は情報要求に基づいて2段階のデコーディングを必要とするが、これは下記のとおりである。

* 3ビットマクロブロックタイプを有するデコーディングMCBPC

* マクロブロックタイプ、BPフラグ及びピクチャタイプに基づいたマクロブロックタイプブロックアップVLCテーブルでマクロブロックタイプのフォーマットは下記のとおりである。

【0205】

【表63】

H. 263のMCBPCフォーマット

ビット4～2	ビット1	ビット0
MBタイプ	CR	CB

【0207】

【表65】

H. 261のマクロブロックタイプフォーマット

ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
FIL	Q	MVD	予約	CBP	INTARA

表から、3ビットマクロブロックタイプだけでなく、2ビットクロマパターンを得る。ここにおいて、マクロブロックタイプは、0～4までの範囲を有する3ビット値である。上述のごとく、細部的なマクロブロックタイプ*

*の種類は、第2段階でデコーディングされる。デコーディングルックアップテーブルは下記のとおりである。

【0208】

【表66】

H. 263のマクロブロックタイプデコーディングルックアップテーブル

PT	MB TYPE	PB	M4	Q	MF	MODEB	P(Y)	INTRA
INTER	3	0	0	0	0	0	1	1
INTER	4	0	0	1	0	0	1	1
INTER	0	0	0	0	1	0	1	0
INTER	1	0	0	1	1	0	1	0
INTER	2	0	1	0	1	0	1	0
INTER	3	0	0	0	0	0	1	1
INTER	4	0	0	1	0	0	1	1
INTER	0	1	0	0	1	1	1	0
INTER	1	1	0	1	1	1	1	0
INTER	2	1	1	0	1	1	1	0
INTER	3	1	0	0	1	1	1	1
INTER	4	1	1	1	1	1	1	1

【0209】1. 2. 7 DCT DC サイズ

この節は、テーブル12、13/MPEG-2、テーブル2-B. 5/MPEG-1を取扱う。VLC構造によって‘1’カウントはここで‘0’カウントの代わりに使用される

ROMテーブルフォーマット：ビット10～6：上位データ：クロマ；ビット5～0：下位データ：輝度。ビット11とビット5は予約されている。

1. 2. 8 CBPY

50 この節は、テーブル9/H. 263を取扱う。このテー

ブルは二セットのデータを含むが、一つはインターピクチャーに関するもので、もう一つはイントラピクチャーに関する。一セットの値は、他セットの値を反転させたものであって、ROMに一セットのデータが貯蔵できるようにする。ここにおいて、イントラデータがROMに位置する。1つの4ビット値がCBPY値を表わすことに使用される。

ROMテーブルフォーマット：ビット9～6：上位データ；ビット3～0：下位データ。ビット11～10とビット5～4とは予約されている。

1. 2. 9 デュアルプライム (dual prime) 及びモード

この節はテーブル11/MPEG-2及びテーブル7/H. 263を取扱う。この2個のテーブルは、非常に簡*

一般的なエンコーディングフォーマット

ビット11～ビット8	ビット7～ビット5	ビット4～ビット0
0/1カウンタ	VLCコードサイズ	VLCコード

【0211】

【表68】

テーブル/H. 263エンコーディングフォーマット

ビット11～ビット9	ビット8～ビット6	ビット5～ビット0
0/1カウンタ	VLCコードサイズ	VLCコード

【0212】前記のテーブルで、VLCコードサイズは最初/最後‘1’を後続くVLCコードのサイズである。VLCコードは、最初/最後‘1’を後続くVLCコードである。‘0’カウンタの場合、最初の‘1’を後続くVLCコードが抽出され、そうでない場合VLCコードは最後の‘1’を後続くビットから抽出されなければならない。エンコーディングで‘1’カウンタの適用は、デコーディングでのそれとは異なる。‘1’カウンタは‘1’カウンタフラグがアドレス発生器によりイネーブルされる場合にのみ適用される。従って、VLCのMSBが1であるが‘1’カウンタフラグがローの場合、ROMテーブルの‘0’/‘1’カウンタ部分は0となり、これは‘0’カウンタが適用されることを意味する。次の例は、エンコーディングに対するすべての可能な場合を取扱う。

例1：VLC=0000011001, one_count_enable=0

一般的な場合に対する結果：0101 100 01001

テーブル12/H. 263に対する結果：101 100 0010 01

例2：VLC=11001, one_count_enable=0

一般的な場合に対する結果：0000 100 0101

テーブル12/H. 263に対する結果：000 100 0010 01

*単で小さくて、直接デコーディングされ得る。

1. 3 エンコーディング

デコーディング節と同様に、エンコーディング過程は‘0’/‘1’カウンタ概念を使用する。ROMテーブルは、‘0’/‘1’カウンタ、‘0’または‘1’カウンタに対して最初1に後続くコードのサイズ及び、最初/最後‘1’に後続くVLCコードに対する情報を含む。このフォーマットによると、ROMテーブルのサイズは、テーブル12/H. 263において特殊エンコーディングで解決される4つを除いては、12ビットに制限され得る。フォーマットは下記のとおりである。

【0210】

【表67】

例3：VLC=11001, one_count_enable=1

一般的な場合に対する結果：0010 011 00001

テーブル12/H. 263に対する結果

一般的なアドレスは、オフセットと入力値の加算によって発生される。

【0213】1. 3. 1 テーブル14/MPEG-2

このテーブルはテーブル2-B. 5c/MPEG-1及びテーブル5/H. 261と同一である。このエンコーディングは‘RUN’、‘FIRST DC’、‘ESCAPE’、‘END OF BLOCK’入力を処理する。エンコーディング結果：アドレスを発生するためにレベルまたはランと共に加算されるように印加されるオフセットアドレス

1. 3. 2 テーブル15/MPEG-2

二テーブルが同一なラン、レベル及びVLCコードを有するので、このテーブルの大部分はテーブル14/MPEG-2を共有する。いくつかの特殊な場合において、‘1’カウンタが適用される。このエンコーディングは‘RUN’、‘LEVEL’、‘FIRST DC’、‘ESCAPE’、‘END OF BLOCK’入力を処理する。

エンコーディング結果：オフセットアドレス及び‘1’

50 カウンタ指示者

1. 3. 3 テーブル12/H. 263

上述のごとく、このテーブルは非常に特殊である。これを取扱うために他のフォーマットを使用する。不幸にも、VLCコードを示すことに12ビットが使用できない幾つかの例外がある。その例外はテーブル9に示したとおりである。この例外はROMテーブルを使用せず、特殊にエンコーディングされ得る。

【0214】

【表69】

12/H. 263でエンコーディングの例外

最 後	ラン	レベル
0	0	10
0	0	11
1	0	3
1	1	2

【0215】エンコードは‘RUN’及び‘ESCAPE’入力を処理する。

エンコード結果：アドレスを発生するために、レベルまたはランと共に加算されるように印加されるオフセットアドレス

1. 3. 4 モーションコード/マクロブロック増加分

この節は、テーブル1/MPEG-2、テーブル10/MPEG-2、テーブル2-B. 1/MPEG-1、テーブル2-B. 4/MPEG-1、テーブル1/H. 261、テーブル3/H. 261及びテーブル10/H. 263を取扱う。デコーディング部分で説明したとおり、このすべてのテーブルに対して、1つのROMテーブルと1つのFSMとを共有することができる。ROMテーブルから得られるVLCコードは、完全なVLCコードを作るために符号/偶数ビットと結合しなければならない。従って、このエンコーディングFSMで処理する入力値は、そのLSBがフラクションビット(fraction bit)であるモーションコードに対する絶対値と、1ビットの右側にシフトされたマクロブロックアドレス増加分である。

【0216】エンコーディングは‘STUFFING’及び‘ESCAPE’を処理する。

1. 3. 5 マクロブロックパターン

この節はテーブル9/MPEG-2、テーブル2-B. 3/MPEG-1を取扱う。アドレスはオフセットとパターン値を加算した値である。

1. 3. 6 マクロブロックタイプ

この節は、テーブル2、3、4/MPEG-2、テーブル2-B. 2/MPEG-1を取扱う。

1. 3. 7 テーブル3、4/H. 263 (MCBP C)

ピクチャータイプ、マクロブロックタイプ及びスタッフィング(stuffing)フラグに対する情報が、ROMテーブルアドレスオフセットを発生させるために提供される。アドレスはオフセットアドレスとCBPCの和である。

1. 3. 8 テーブル2/H. 261 (MTYPE)

アドレス発生器が非常に複雑で、具現に対して考慮する価値がない。

10 1. 3. 9 CBPY

デコーディング部分に述べたように、イントラピクチャーデータのみをエンコーディングする。ピクチャータイプがインターピクチャーである場合、データはまず反転されなければならない。アドレスはオフセットとCBPY値を加算した値である。

1. 3. 10 DCT DCサイズ

この節は、テーブル12、13/MPEG-2、テーブル2-B. 5/MPEG-1を取扱う。輝度及びクロマに対する数個のVLCコードが同一であるので、これに対して数個のROMテーブルを共有する。オフセットアドレスを発生させるために、クロマフラグ及び数個のビット値が使用される。オフセットと実際値を加算することによってROMアドレスを得ることができる。

【0217】1. 3. 11 デュアルプライム(dual prime)及びモード

この節は、テーブル11/MPEG-2及びテーブル7/H. 263を取扱う。この2個のテーブルは非常に簡単で小さくて、直接エンコーディングされ得る。

2. 0 ハードウェア説明

VLCエンコーディング/デコーディングに対するハードウェアは、‘VLC’ブロックに含まれる。このブロックは3個のサブブロックを含む。このブロックはROMテーブルアドレスまたはデコーディング/エンコーディングデータを発生させるために適用される。‘VLC-DEC’はVLCをデコーディングし、ROMテーブルアドレスを発生させるために使用される。‘VLC-ENC’はVLCをエンコーディングするためのブロックであり、ROMテーブルアドレスまたはH. 263のTCOEFテーブルのための特殊エンコーディングを発生させる。‘LOOKUP’はROMテーブル値または特殊エンコーディング値に基づいてVLCデータを出力する。

2. 1 VLCデコーディングアドレス発生器

VLC-DECの核心は、デコーディングFSMである。このFSMは入力情報をデコーディングしアドレス発生を制御する。FSMの入力及び定義は下記のとおりである。

*ZERO_ONE Count (15ビット) : 0/1 カウント値を提供する。

50 *ZERO_ONE Count (4ビット) : 0/1

カウント値を提供する。2個の相互に異なるビットカウント信号を使用する目的は、入力データ共有により、ゲート注文者(gate customer)を減少させるためである。大部の場合、15ビットが使用される。

*ONE Count enable (1ビット) : *

VLC_DEC_FSMテーブルタイプフォーマット

*モード (9ビット) : 動作モード

ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
DCTサイズ	MB INC/MC	MBパターン	CBPY	MB TYPE	DC COEF

【0219】

VLC_DEC FSMモードフォーマット

ビット8 ビット7 ビット6 ビット5 ビット4 ビット3 ビット2 ビット1 ビット0
H.263 仕様 ビクチャタイプ クロマ 第1 DC テーブル15 MB INC

仕様及びビクチャタイプに対する定義は、ピン定義で説明する。ハードウェアを簡素化し、ROMアクセス時間を確保するために、特殊なアルゴリズムがこのデコードROMテーブルアドレスを発生させるために使用される。この過程は下記のとおりである。

【0220】段階1：オフセットアドレス(OFFSET)を発生させる。

段階2：4ビットシフト量(MASK_SHFT)を発生させ、この量と共に右側シフト16ビットFIFO_DATAを※

Address=OFFSET | (BITREVERSE(Bit(3~0) of (FIFO_DATA)>>MASK_SHFT))

& MASK)

FSMの出力は下記のとおりである。

*MASK (4ビット) : マスタデータ

*OFFSET (9ビット) : ROMテーブルオフセットアドレス

*MASK_SHFT (4ビット) : シフト量 ★

VLC_DECの特殊フラグの定義

位置	値	定義
ビット2		VLCのLSBを圧縮
ビット1~0	01	ESCAPE
	10	ブロック/スタッフィング終了
	11	H.261のスタートコード

【0222】*VALID_VLC (1ビット) : 有効VLCコードフラグ

*HIGH_DATA_INDICATOR (1ビット) : ROMデータの中の上位6ビットを抽出する。

入力ピン:

* '1' カウント指示者

*テーブルタイプ (6ビット) : テーブルタイプ

【0218】

【表70】

※発生させる。以降、4個の最下位ビット(FOL_DATA)を抽出する。

20 段階3：段階2から得られる4個ビットを反転させる。
段階4：段階3から得られるデータをマスク(MASK)するために、4ビットマスク信号を発生させる。

段階5：段階4の結果をオフセットアドレスと論理和させる。その結果はROMテーブルアドレスである。この段階を結合させると下記のとおりである。

★*SIZE (5ビット) : VLCサイズ

30 *SPECIAL_FLAG (3ビット) : デコーディングのための余分の情報

【0221】

【表71】

*FOL_DATA (4ビット) : シフトされたFIFO_DATA (上述した段階2参照)

*CNT (4ビット) : 0/1カウント

*ONE_CNT_EN (1ビット) : '1' カウント

50 指示者

*MODE (14ビット) : テーブルタイプ及び他の情報

*【表72】

VLC__DECでMODEフォーマット

定義は下記のとおりである。

【0223】

*

ビット13	ビット12	ビット11	ビット10	ビット9	ビット8	ビット7
DCT SIZE	MC	MB INC	MB パターン	CBPY	MB タイプ	DCT COEF

ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
仕様		ピクチャタイプ		第1DC		テーブル15

【0224】仕様：00=MPEG-1；01=MPEG-2；10=H. 261；11=H. 263；
ピクチャタイプ：00=予約；01=イントラ；10=予測；11=両方向；

*FIFO_DATA (16ビット) : データはVLCを含む。

出力ピン：

*ROM_ADR (10ビット) : ROMテーブルアドレス

*MASK_SHFT (4ビット) : FIFO_DATAに対するシフト量 (上述した段階2参照)

*SIZE (5ビット) : VLCサイズ

*SPECIAL_0 (3ビット) : 特殊フラグ (FSM出力参照)

*VALID_VLC (1ビット) : 有効VLCフラグ

*HIGH_DATA (1ビット) : 偶数フラグの符号として、VLCのLSBの抽出指示者

*FULL_DATA (1ビット) : DCT係数デコーディング時に、ハイの完全な12ビットデータ構造

*TABLE (6ビット) : FSM入力に定義される。

*T_MODE (9ビット) : FSM入力にMODEに定義される。

2.2 VLC_ENC

VLCエンコーディングコア部分でのように、VLC_ENCは可変長さコードをエンコーディングする。この部分の出力は、ROMテーブルアドレスまたはVLCの特殊エンコーディングである。1. 0節の説明のとおり、エンコーディングデータ構造は、H. 263でTCOEFのいくつかの特殊な場合を除いては、12ビットデータフォーマットに従う。たとえ、10ビット加算器がROMテーブルアドレスを発生させるために使用されるが、ハードウェア観点から見ると、VLC_DEC部分より更に簡単である。VLC_DECと同様に、この部分の核心はVLC_ENCというFSMである。他のFSM、ENC_SPは、特殊エンコーディングのために使用される。FSM VLC_ENCの入力信号は、

この部分の入力ピンと同一である。

*LAST (1ビット) : H. 263のTCOEFに対するLAST値

*RUN/VALUE (6ビット) : DCT係数テーブルがエンコーディングの途中であれば、この入力RUNを意味し、そうでない場合一般的な値、すなわちパターンを意味する。

*LEVEL (6ビット) : DCT係数レベル

*SPECIAL_FLAG (2ビット) : VLC_DEC部分で定義された特殊フラグ

*TABLE (6ビット) : VLC_DECと同一

*MODE (9ビット) : VLC_DECと同一

ROMアドレス発生は非常に簡単である。FSMはアドレスを発生させるために値 (ラン) またはレベルまたは0に加えられるオフセットアドレスを提供する。このVLCが同一サイズと '0' カウントを有するので、特殊エンコーディングのためには、出力はコードで復元される2個の最下位ビットである。

【0225】出力ピンは下記のとおりである。

*ONE_CNT_FLG (1ビット) : VLC構造が '1' カウントを使用することを知らせる。

*SIGN_EN_BIT : VLC構造が符号/偶数ビットをVLC_LSBに置くことを知らせる。

*SPECIAL_ENCODE (1ビット) : 特殊エンコーディングフラグ

*VLC (2ビット) 特殊エンコーディングされたVLCコードLSB

*ADR_A (16ビット) : オフセットアドレス。上位6ビットは0である。

*ADR_B (16ビット) : アドレスのまた他の部分。上位10ビットは常に0である。

2.3 ルックアップ

この節は、VLCデータのエンコーディング/デコーディングを提供する。このブロックは下記のような状況で処理する。

*規則的な12ビットエンコーディング/デコーディン

グROMテーブル値出力

*ビット上位/下位デコーディングデータ出力

*特殊エンコーディングデータの復元

要求した通り、出力データ0で満たされる。

入力ピン:

*D_ADR (10ビット): ROMアドレスをデコーディングする。

*E_ADR (10ビット): ROMアドレスをエンコーディングする。

*ENCODE (1ビット): 1:エンコーディング; 0:デコーディング

*HIGH (1ビット): 上位6ビットフラグを抽出する。

*ENABLE (1ビット): 完全な12ビットデータフラグ

*VLC (2ビット): 特殊エンコーディングコード

*SPECIAL_ENCODE (1ビット): 特殊エンコーディングコード

出力ピン:

LOOKUP (16ビット): VLCコード

[0226]

【発明の効果】以上述べたように、本発明はビットストリーム処理器では、多様なビットストリームが実時間的に同時にエンコーディングまたはデコーディングされるように文脈を貯蔵することができるので、多重データストリームを同時に処理することができる。また、ビットストリーム処理器が単一算術命令またはブール命令を遂行するためにプログラムされないようにすることにより、ビットストリーム処理器が高速で動作できる。

【図面の簡単な説明】

【図1】 本発明によるメディアカードのブロック図である。

【図2】 本発明によるマルチメディア処理器のブロック図である。

【図3】 図2に示された処理器の一部のビットストリーム処理器のブロック図である。

【図4】 本発明によるコンピュータシステムのブロック図である。

【図5】 本発明によるコンピュータシステムのブロック図である。

【図6】 本発明によるコンピュータシステムのブロック図である。

【図7】 図2に示された処理器のファームウェア構造を示す図である。

【図8】 図1のシステムのためのアドレスマップを示す図である。

【図9】 図1のシステムのためのアドレスマップを示す図である。

【図10】 図2に示された処理器のDSPコアを示すブロック図である。

【図11】 図2に示された処理器の一部のベクトル処理器に適用されたパイプラインを示す図である。

【図12】 図11のベクトル処理器の機能的なブロック図である。

【図13】 図11のベクトル処理器において実行データ経路を示す図である。

【図14】 図11のベクトル処理器においてロード及び貯蔵データ経路を示す図である。

【図15】 図2の処理器のキャッシュシステムのブロック図である。

【図16】 図15のキャッシュシステムにおいて命令データキャッシュを示す図である。

【図17】 図2の処理器においてキャッシュ制御ユニットのデータ経路パイプラインを示す図である。

【図18】 図2に図示したシステムにおいてキャッシュ制御ユニットのアドレス処理パイプラインのためのデータ経路を示す図である。

【図19】 図2の処理器においてステートマシンを示す図である。

【図20】 図2の処理器においてステートマシンを示す図である。

【図21】 図2の処理器においてステートマシンを示す図である。

【図22】 図2の処理器においてステートマシンを示す図である。

【図23】 図15のキャッシュシステムで使用されたアドレスフォーマットを示す図である。

【図24】 図2の処理器においてバスを示す図である。

【図25】 図2の処理器において仲裁制御ユニットを示す図である。

【図26】 図2の処理器に対するタイミング図である。

【図27】 図2の処理器に対するタイミング図である。

【図28】 図2の処理器に対するタイミング図である。

【図29】 図2の処理器に対するタイミング図である。

【図30】 図2の処理器においてメモリリクエスト信号を示す図である。

【図31】 図2の処理器においてメモリリクエスト信号を示す図である。

【図32】 図2の処理器においてメモリリクエスト信号を示す図である。

【図33】 図2の処理器においてバス仲裁制御ユニットを示す図である。

【図34】 図2の処理器に対するタイミング図である。

【図35】 図2の処理器に対するタイミング図である。

る。

【図36】 図2の処理器に対するタイミング図である。

【図37】 図2の処理器においてバスインターフェース回路を示す図である。

【図38】 図2の処理器においてバスインターフェース回路を示す図である。

【図39】 図1のシステムに対する仮想フレームバッファ(VFB)を示す図である。

【図40】 図1のシステムに対する仮想フレームバッファ(VFB)を示す図である。 10

【図41】 図1のシステムに対するバスインターフェースを示す図である。

【図42】 図2のシステムに対するメモリコントローラを示す図である。

【図43】 図2のシステムに対するメモリコントローラを示す図である。

【図44】 図2のシステムに対するアドレスコントローラを示す図である。

【図45】 図1のシステムで使用するフォーマットを示す図である 20

【図46】 図1のシステムで使用するフォーマットを示す図である

【図47】 図1のシステムにおいてステートマシンを示す図である

【図48】 図1のシステムに対するデータコントローラのブロック図である。

【図49】 図1のシステムに対するタイミング図である。

【図50】 図1のシステムに対するタイミング図である。 30

【図51】 図1のシステムに対するタイミング図である。

【図52】 図2の処理器において装置インターフェース回路を示す図である。

【図53】 図2の処理器において装置インターフェース回路を示す図である。

【図54】 図1のシステムの各部に対するブロック図である。

【図55】 図1のシステムの各部に対するブロック図 40

【図56】 図1のシステムの各部に対するブロック図である。

【図57】 図1のシステムにおいてレジスタを示す図*

*である。

【図58】 図1のシステムにおいてレジスタを示す図である。

【図59】 図1のシステムにおいてレジスタを示す図である。

【図60】 図1のシステムにおいてフレームバッファ及びビデオウインドを示す図である。

【図61】 図1のシステムに対するタイミング図である。

【図62】 図1のシステムにおいてレジスタを示す図である。

【図63】 図1のシステムに対するタイミング図である。

【図64】 図1のシステムで使用するバッファを示す図である。

【図65】 図1のシステムで使用するバッファを示す図である。

【図66】 図1のシステムで使用するバッファを示す図である。

【符号の説明】

100: メディアカード

105、122: バス

110: マルチメディア処理器

112: D/A変換器

114: CODEC

120: メモリバス

210: スカラー処理器

220: ベクトル処理器

230: キャッシュサブシステム

240: IOBUS

242: タイマー

243: UARTユニット

245: ビットストリーム処理器

250: FBUS

252、255: インターフェース回路

258: コントローラ

290: データ移動器

310: インターフェースユニット

320: SRAM

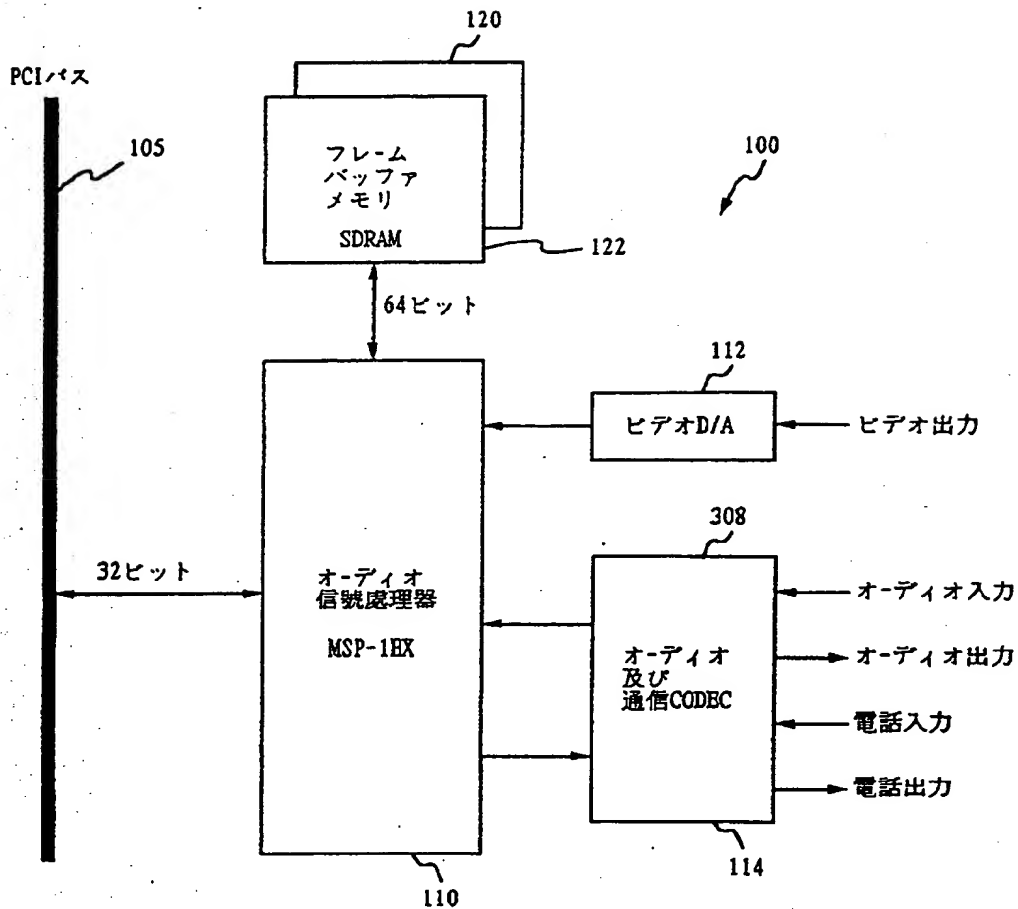
330: VLC FIFOユニット

340: VLC LUT ROM

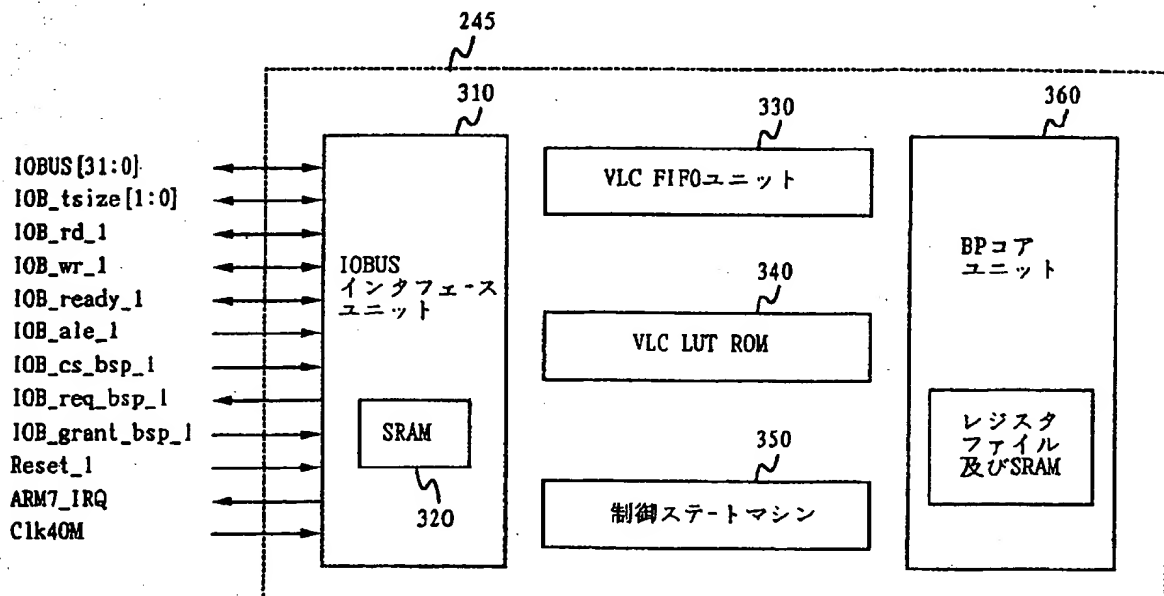
350: 制御ステートマシン

360: BPコアユニット

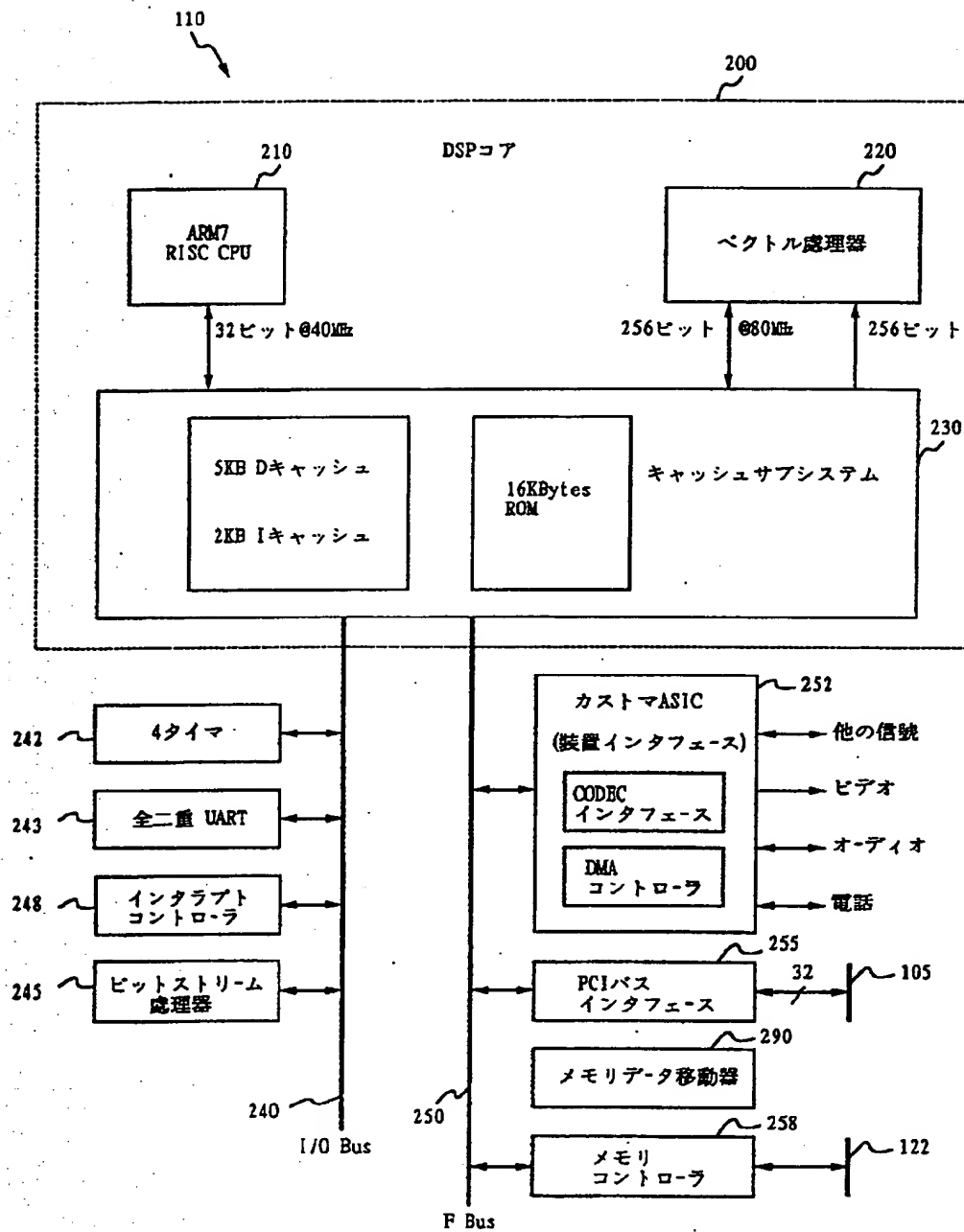
【図1】



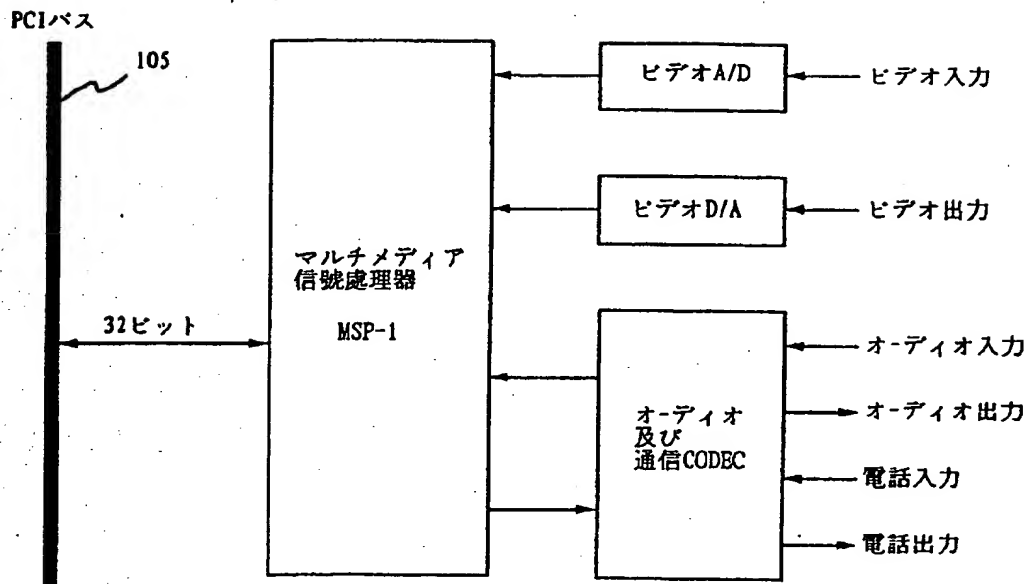
【図3】



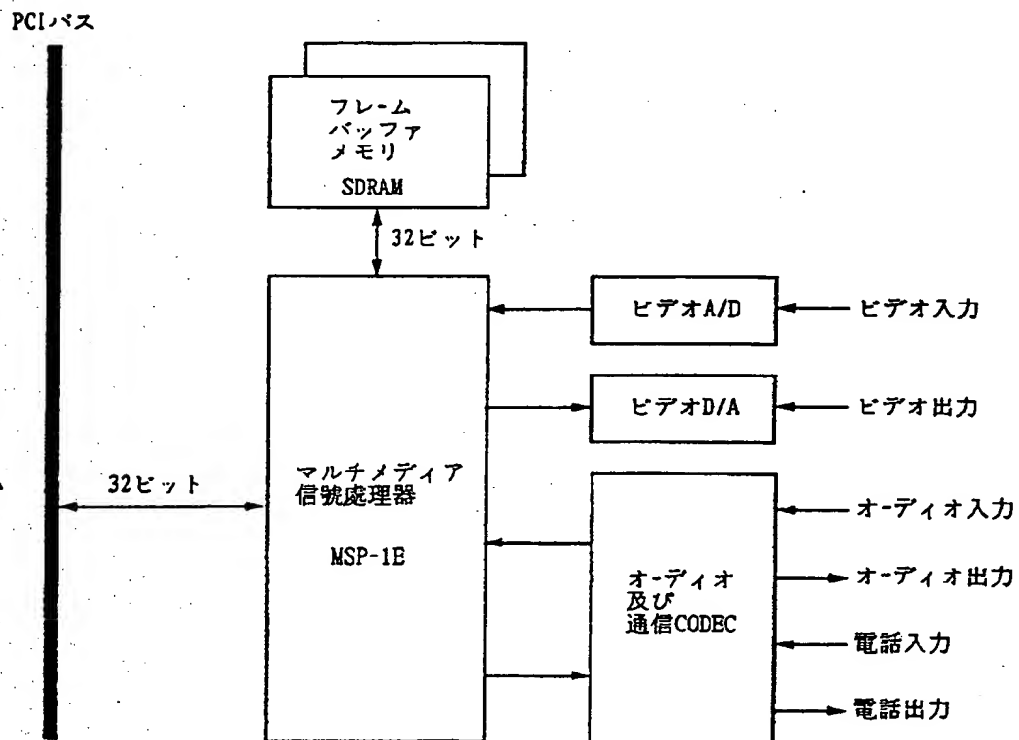
【図2】



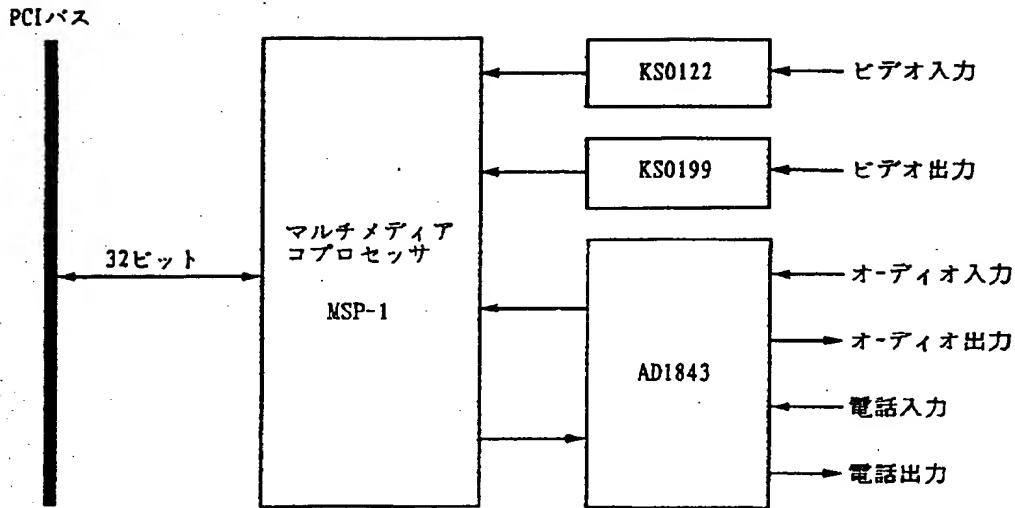
【図4】



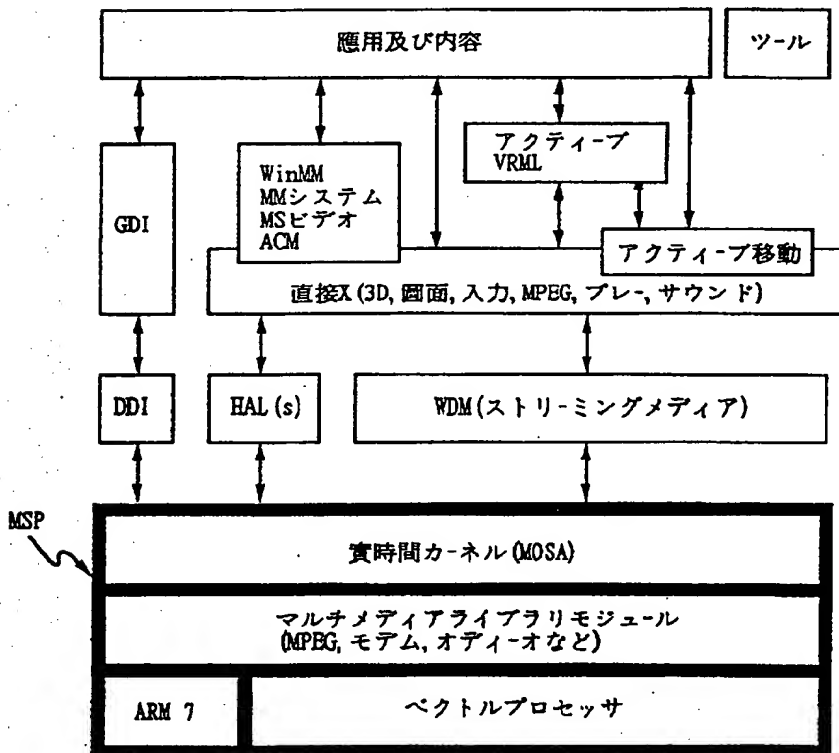
【図5】



【図6】



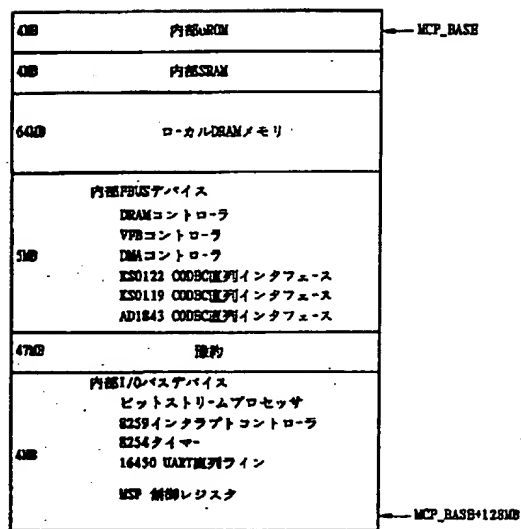
【図7】



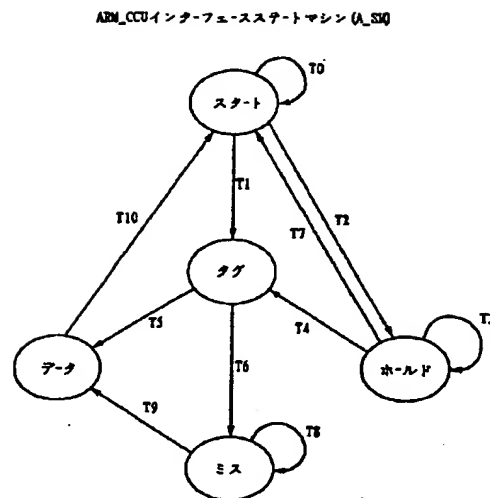
【図8】

4MB	内部uROM
4MB	内部SRAM
64MB	ローカルDRAMメモリ
5MB	内部BUSデバイス DRAMコントローラ VRBコントローラ DMAコントローラ KS0122 CODEC直列インタフェース KS0119 CODEC直列インタフェース AD1843 CODEC直列インタフェース
47MB	稼働
125MB	内部I/Oバスデバイス ビットストリームプロセッサ 8239インタラプトコントローラ 8254タイマ 16450 UART直列ライン MSP 制御レジスタ
2GB	予約
4GB	他のホストデバイス (1GB) (0~2GBホストアドレスにマッピングされる)

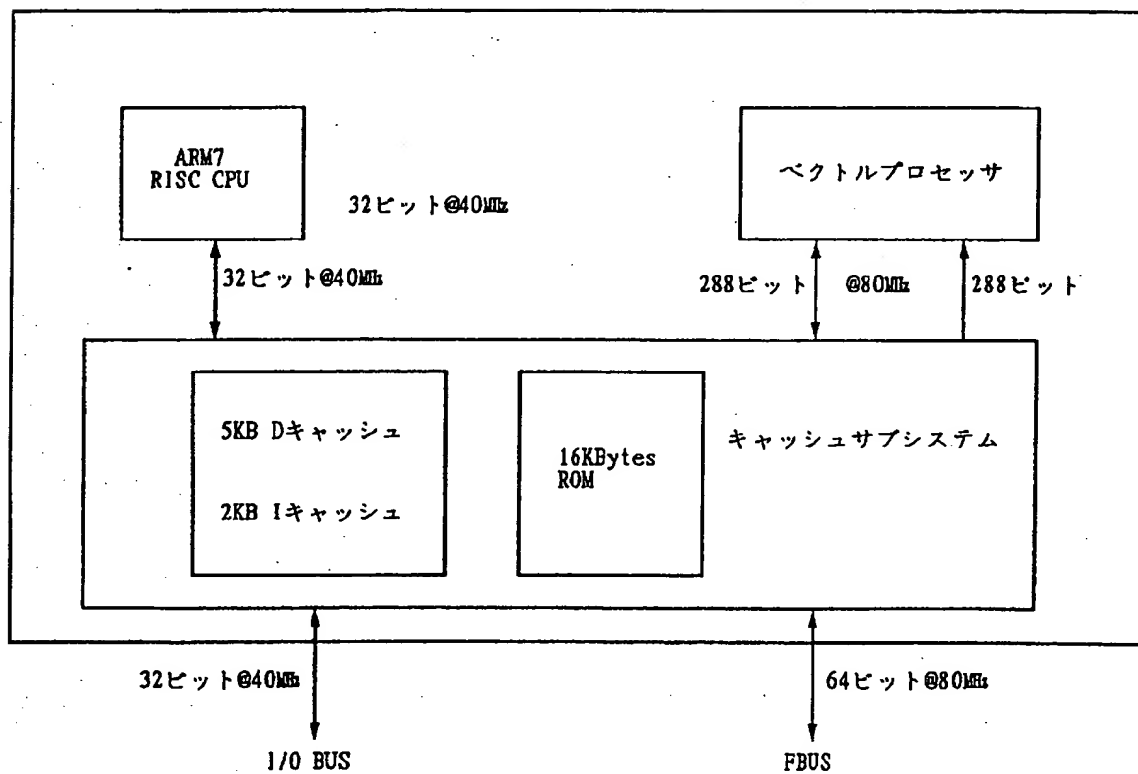
【図9】



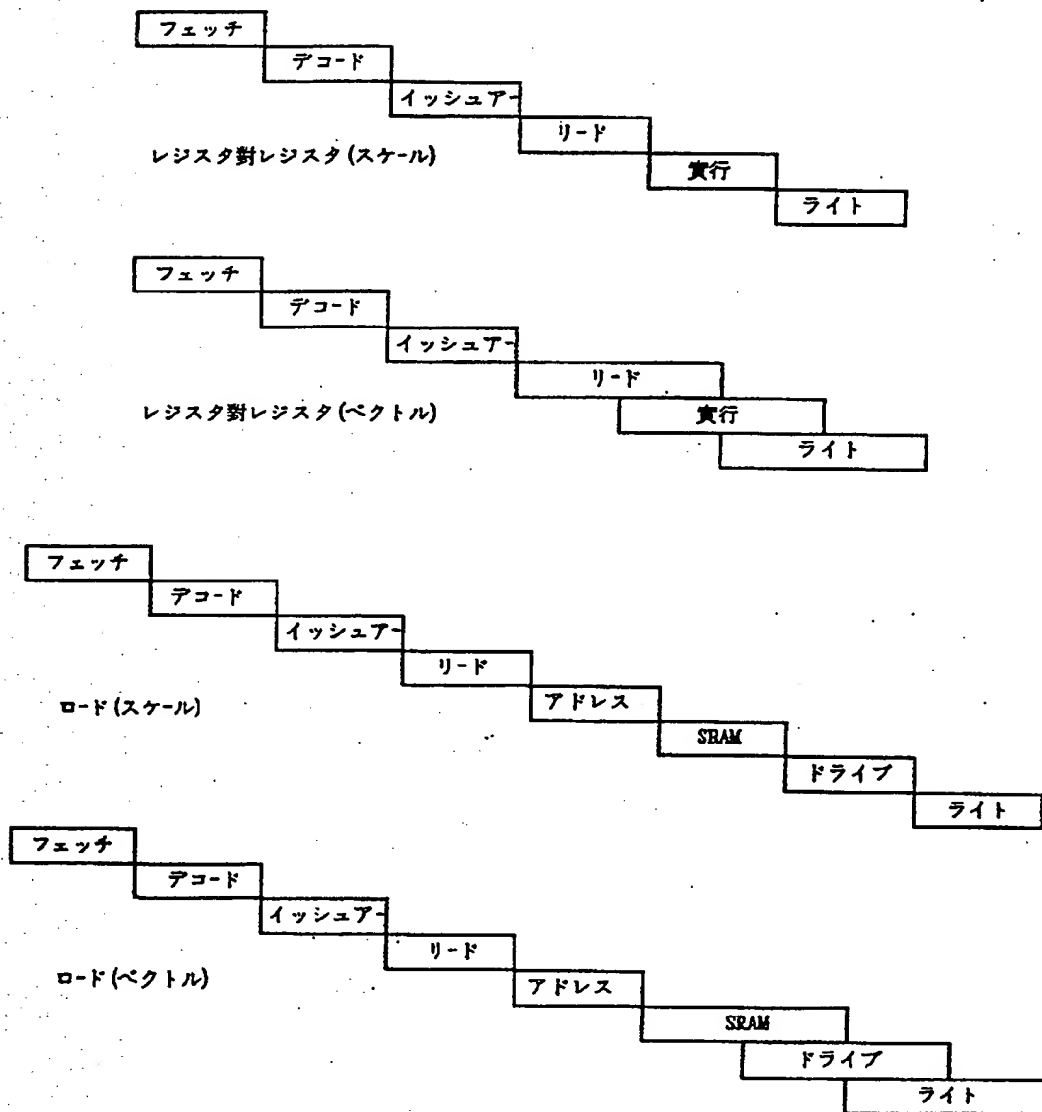
【図19】



【図10】

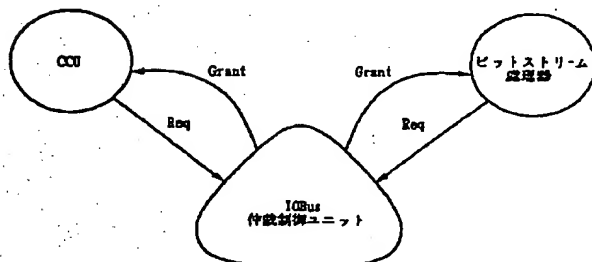


【図11】

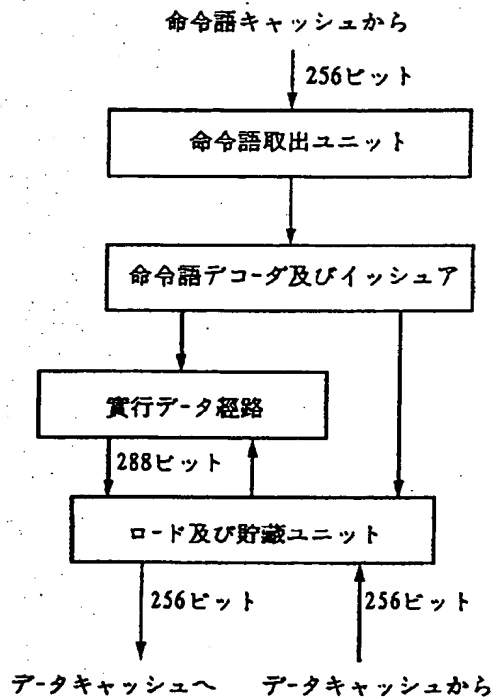


【図25】

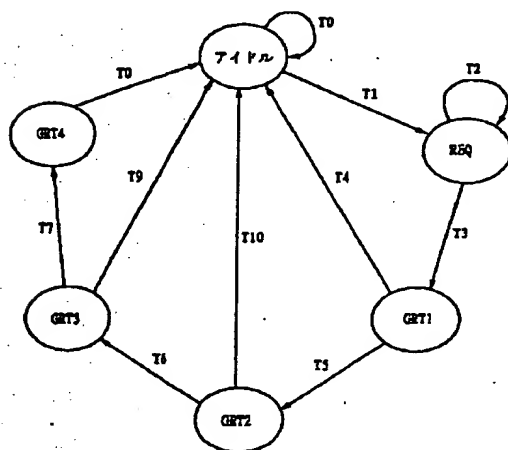
IOBus仲統制御ユニット



【図12】



【図20】



IDLE: アイドル状態

REQ: FPGM アービタに譲渡または記入リクエストを作る

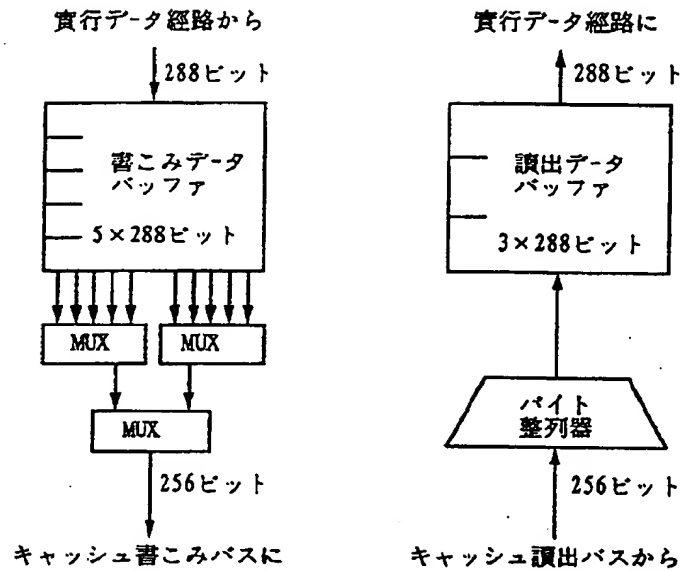
GRT1: 與えられたサイズ8Bより大きい

GRT2: 與えられたサイズ16Bより大きい

GRT3: 與えられたサイズ4Bより大きい

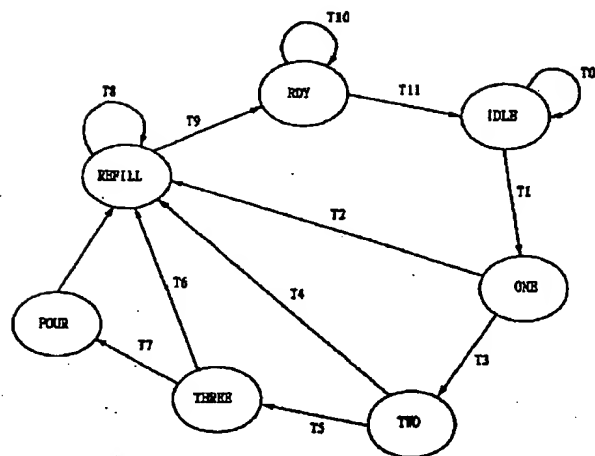
GRT4: 最後のサイクルのための補助データ

【図14】



【図21】

データ受信機ステートマシン (D_S0)



IDLE: アイドル状態

ONE: Fデータ (63:0) から第1番目8Bデータ受信

TWO: Fデータ (63:0) から第2番目8Bデータ受信

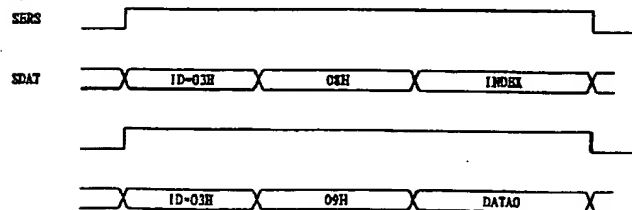
THREE: Fデータ (63:0) から第3番目8Bデータ受信

FOUR: Fデータ (63:0) から第4番目8Bデータ受信

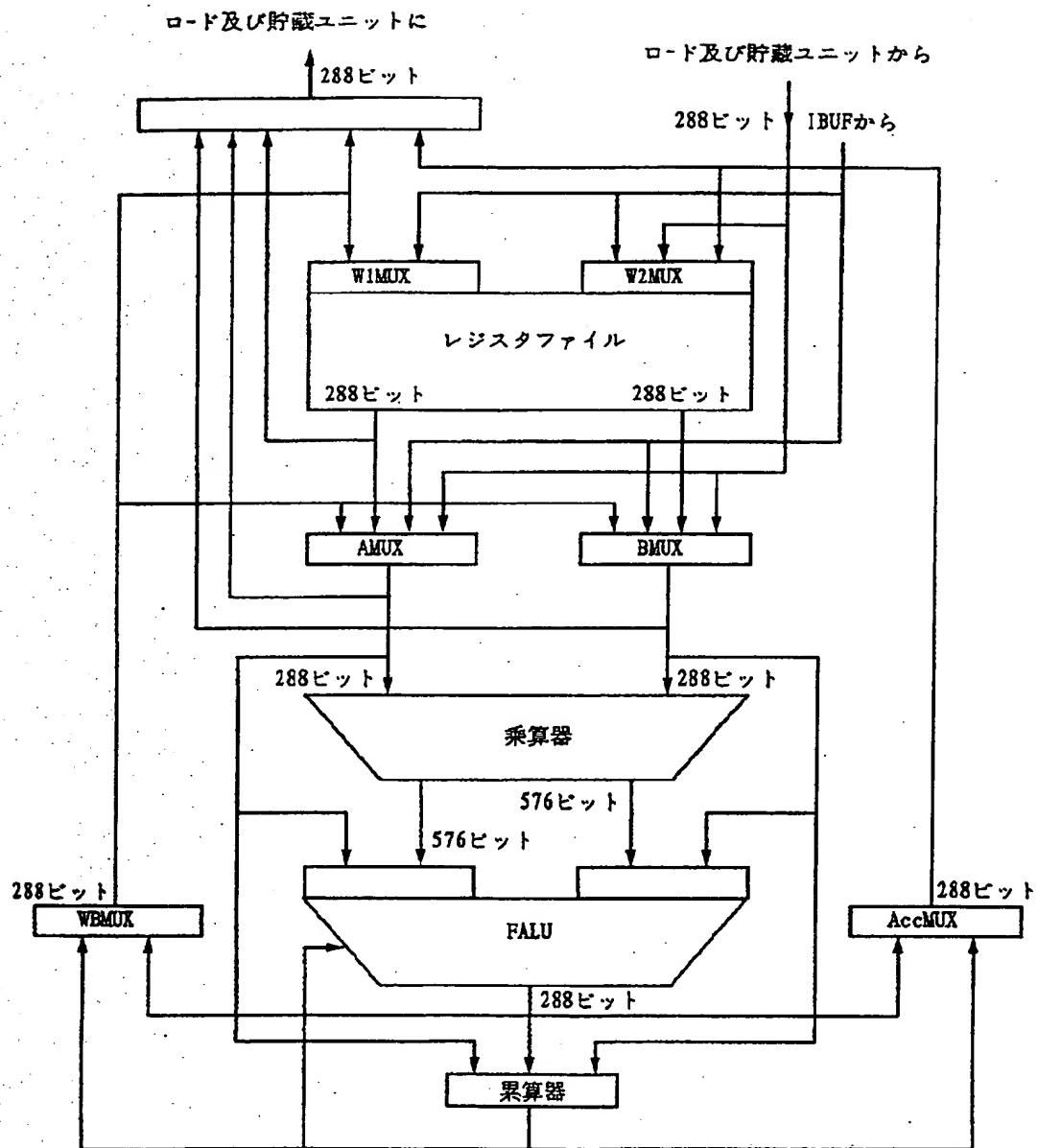
REFILL: リクエストにデータを受領させる前にIDCをリフィール

IDLE: リクエストにデータを受領させる準備

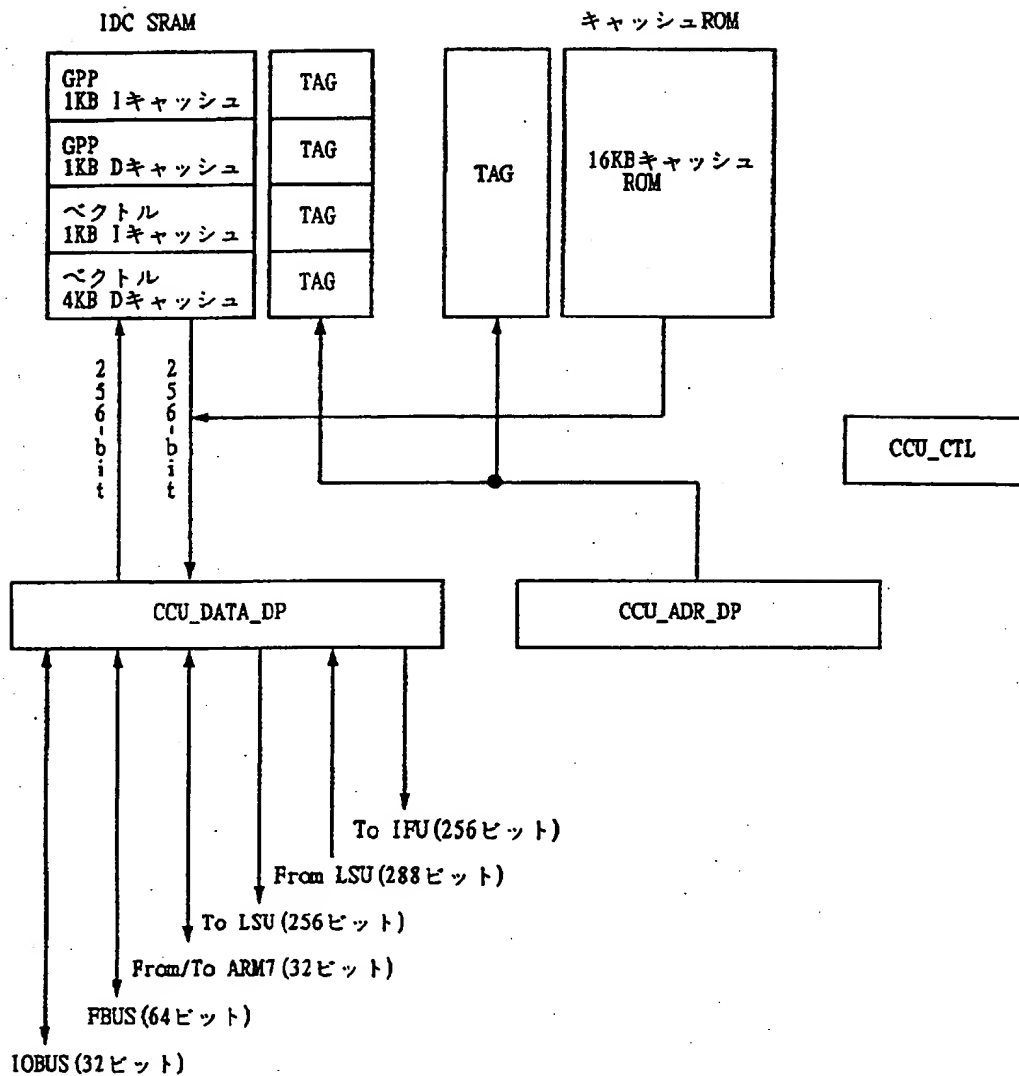
【図61】



【図13】

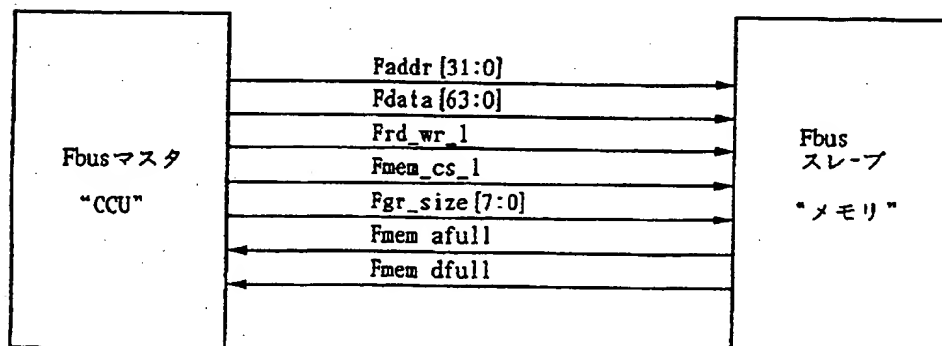


【図15】

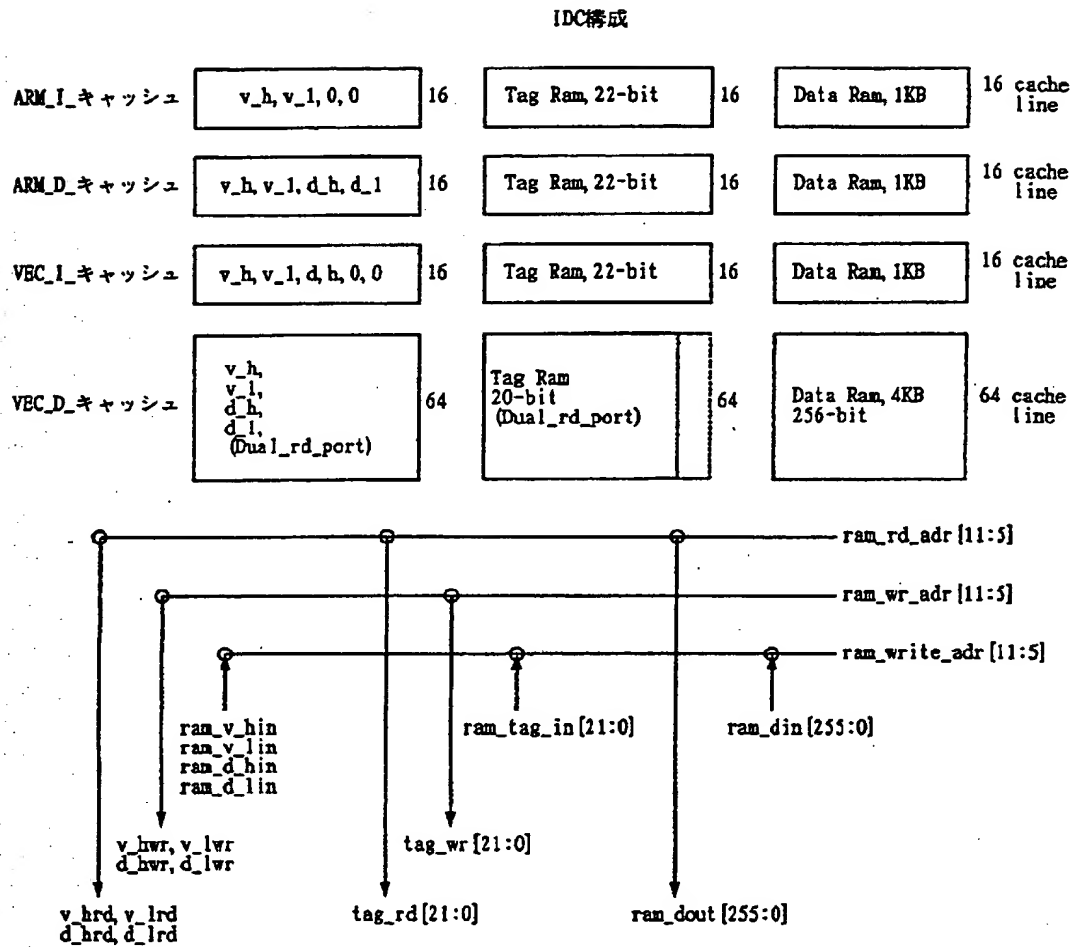


【図31】

メモリ記入リクエストFバスフロー



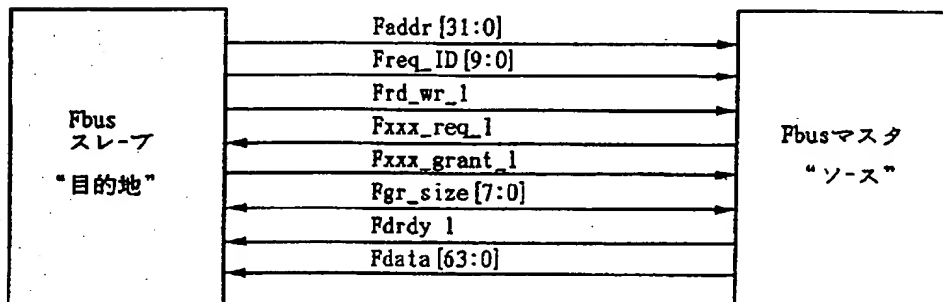
【図16】



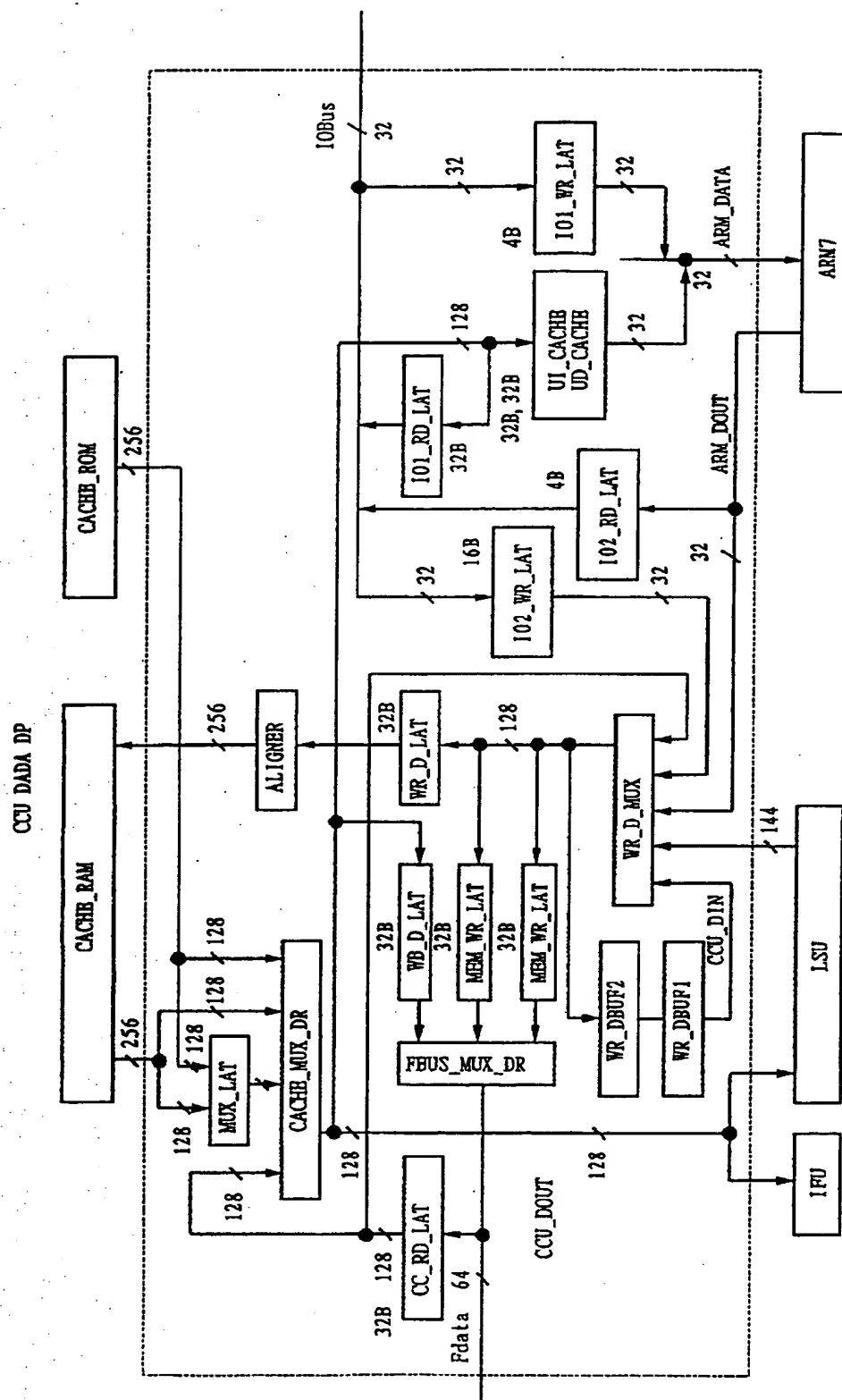
*Note: Tag ram, v/d ram have 1/2 word-line

【図32】

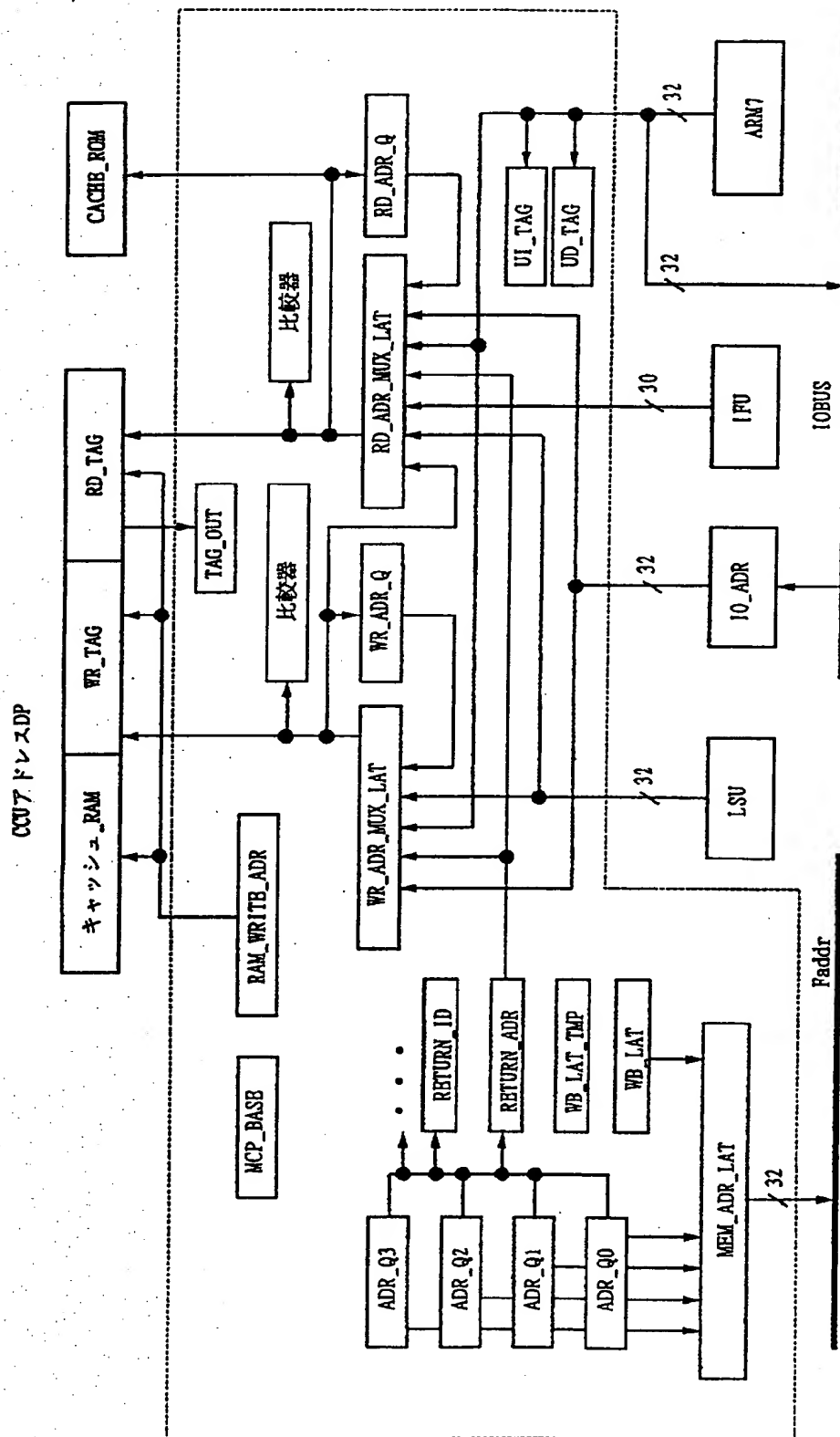
マスタ/スレーブ “非メモリ” リクエスト Fbus フロー



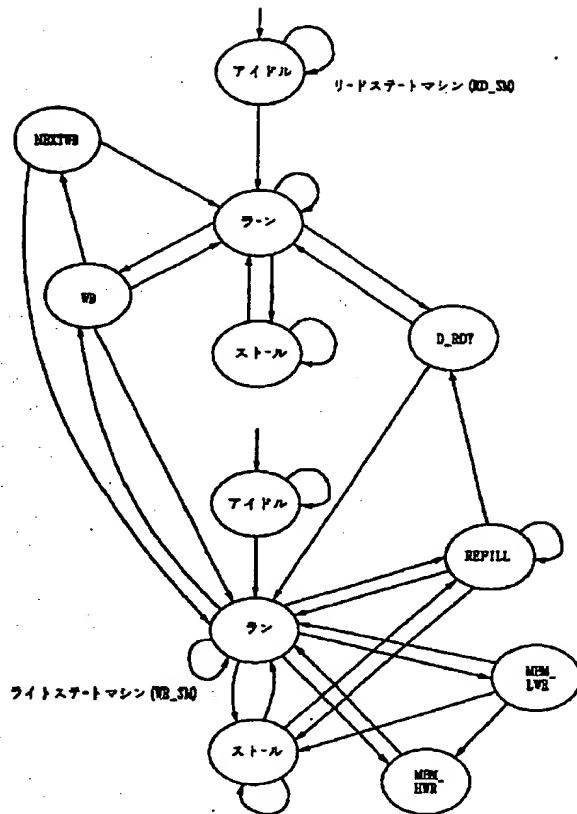
【図 17】



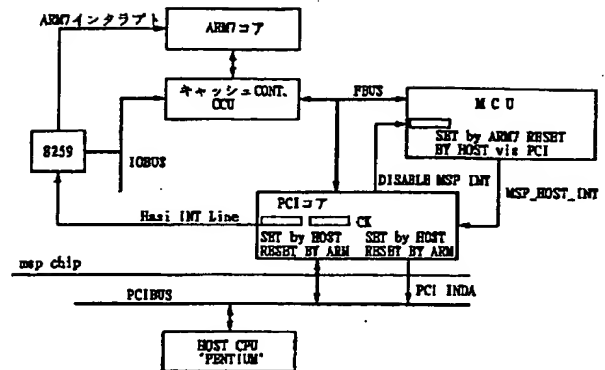
【図 18】



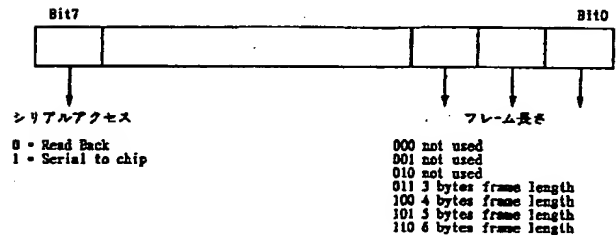
【図22】



【図41】

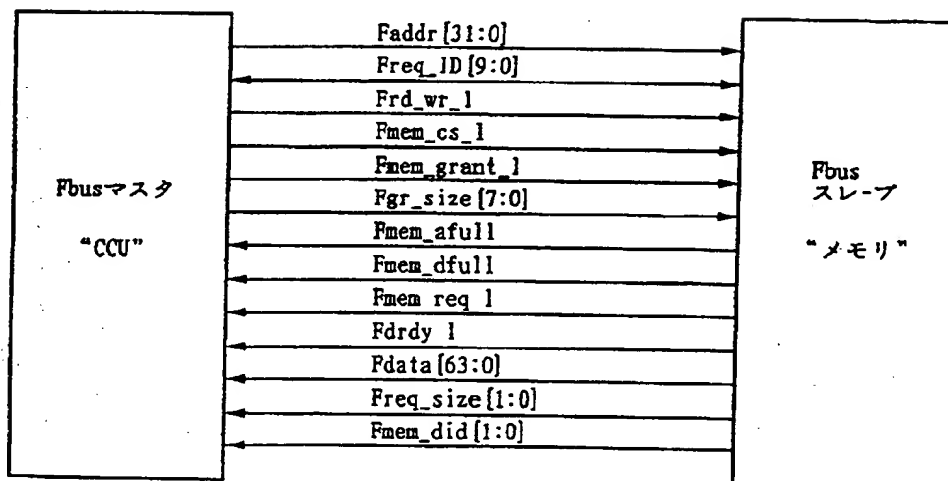


【図57】



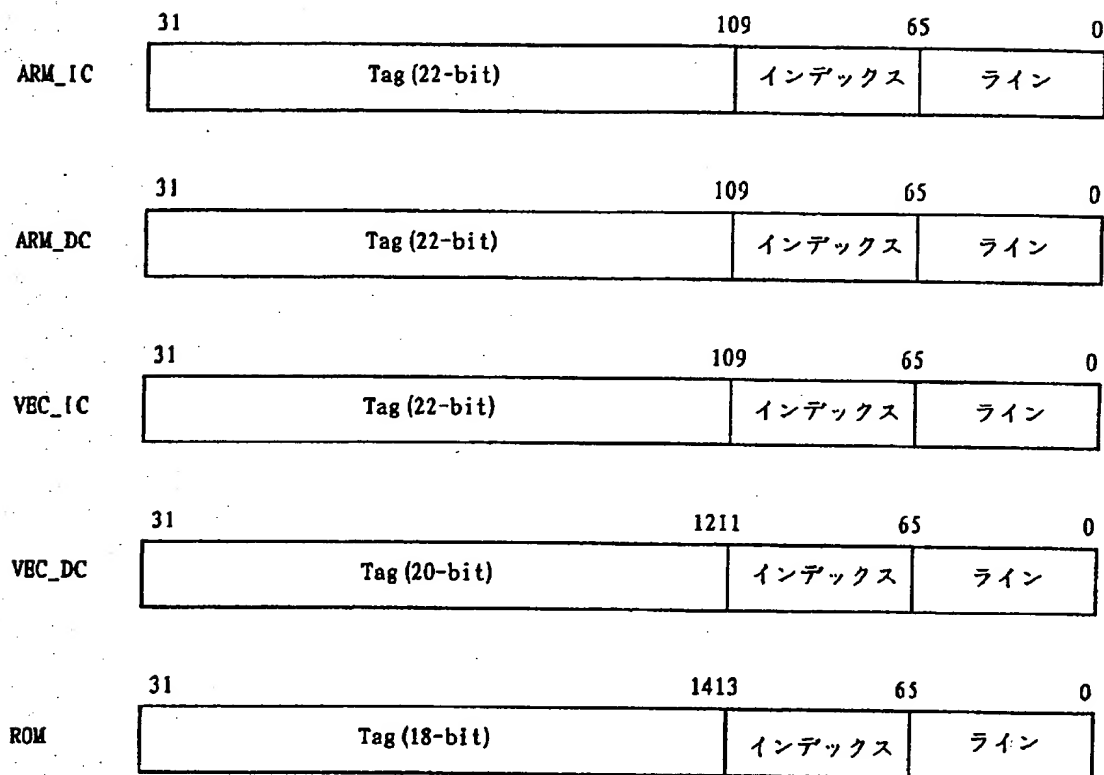
【図30】

メモリ讀取リクエストFバスフロー

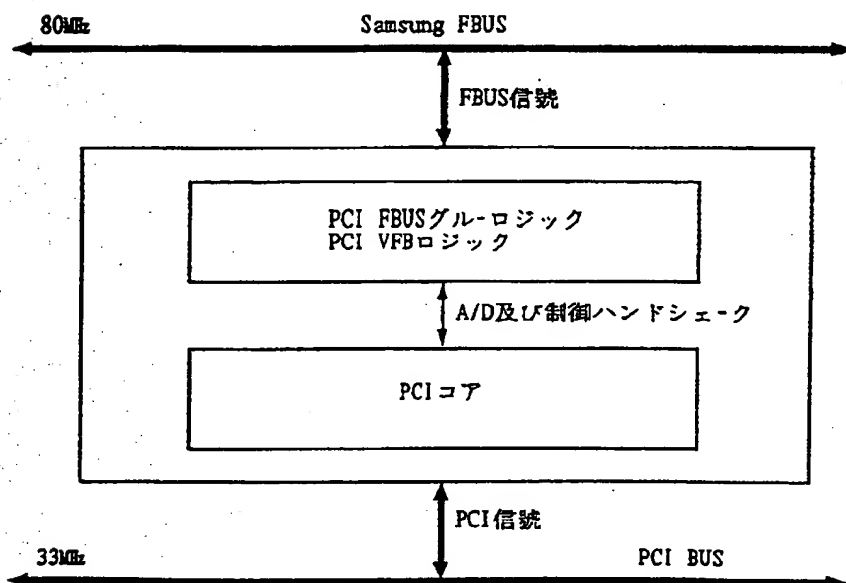


【図23】

IDC and ROMアドレスフォーマット

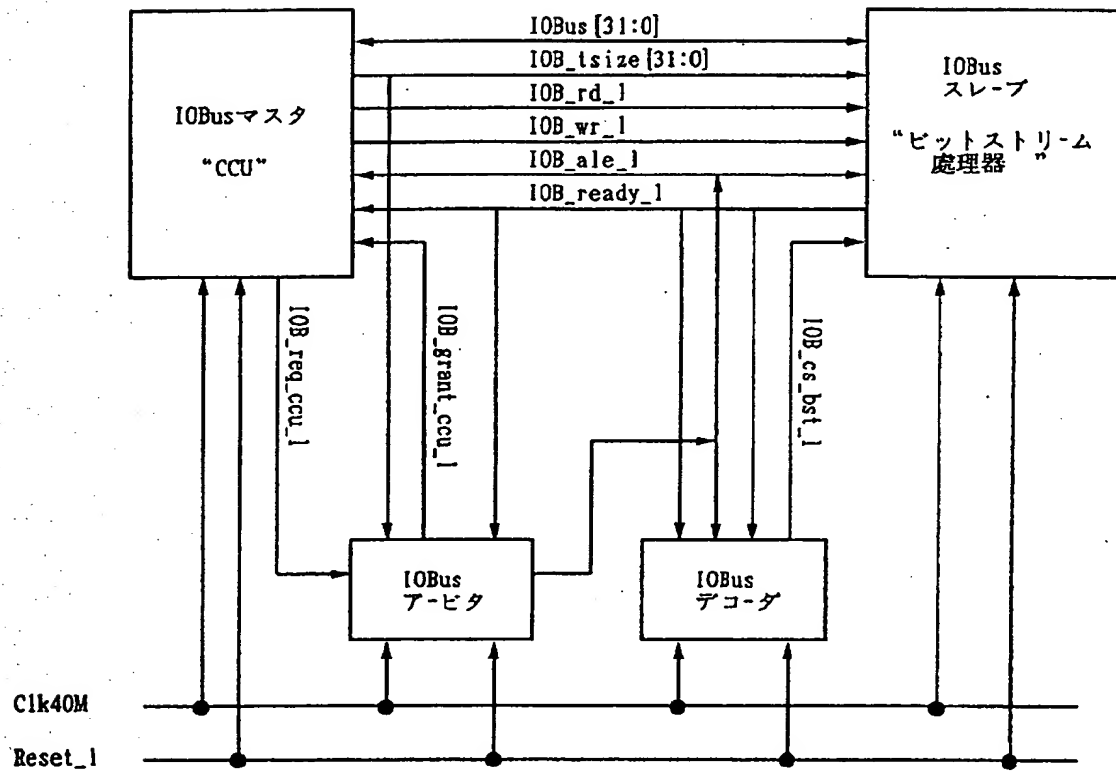


【図37】

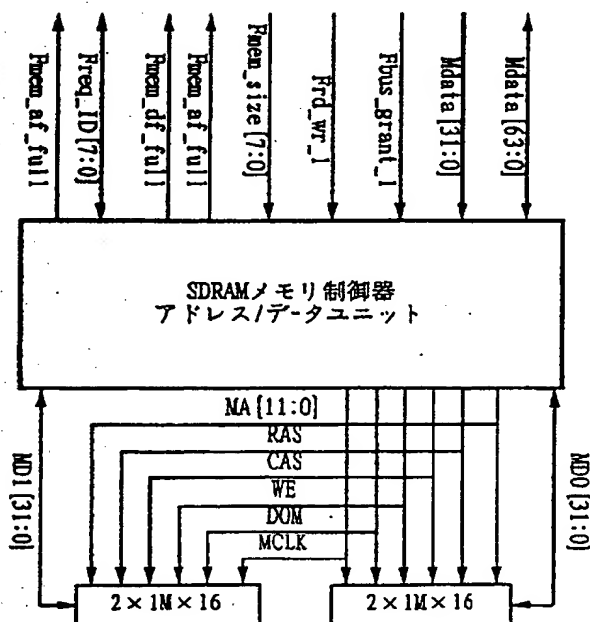


【図24】

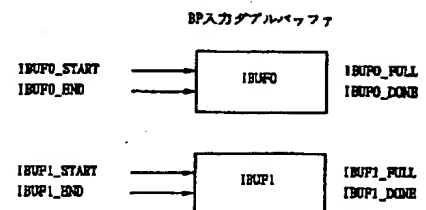
IOBus構造定義



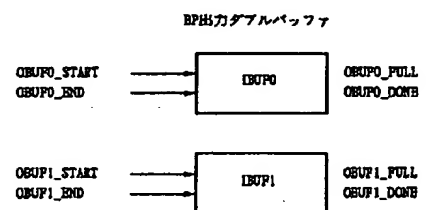
【図42】



【図64】

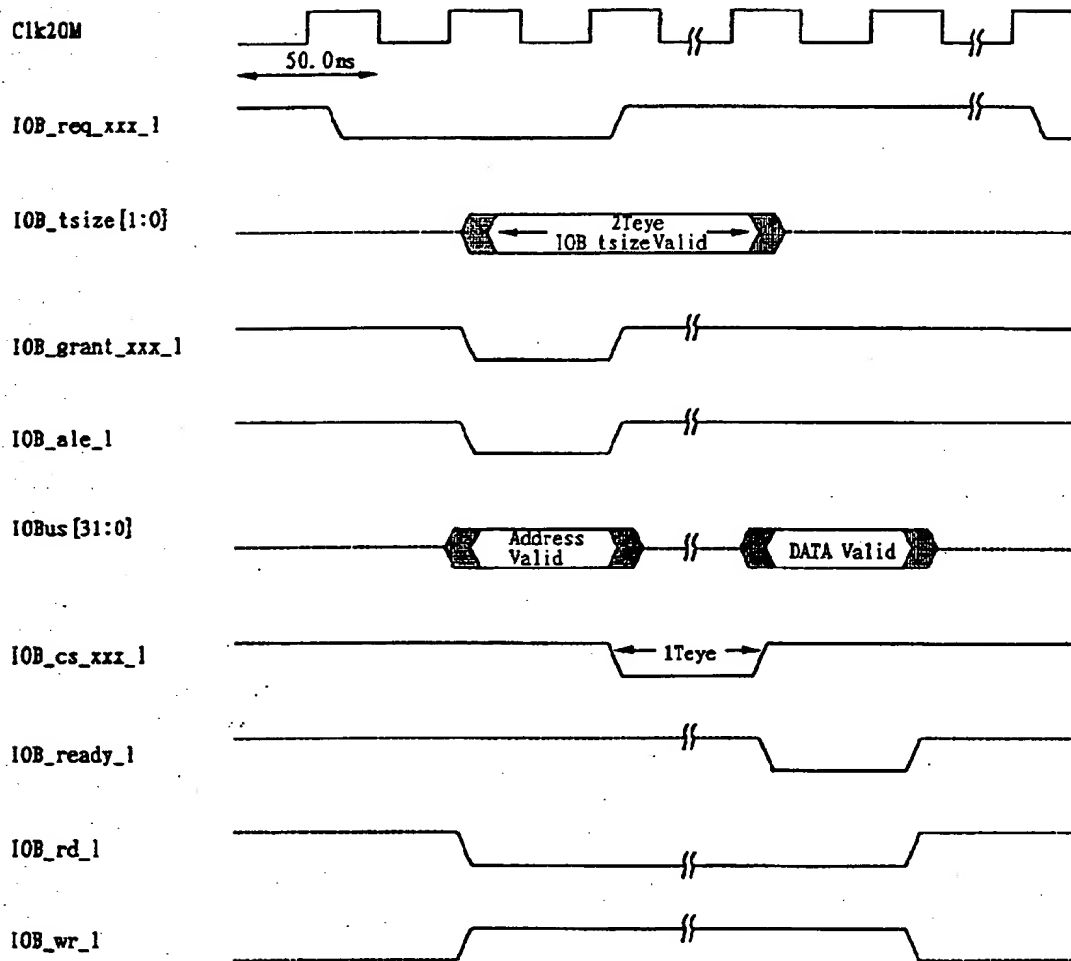


【図65】

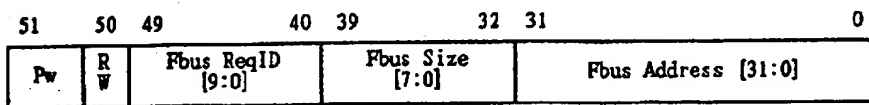


【図26】

IOBusリード (傳送サイズ=1ワード(4bytes))

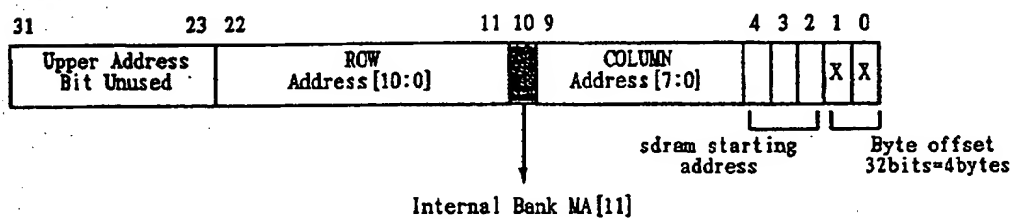


【図45】



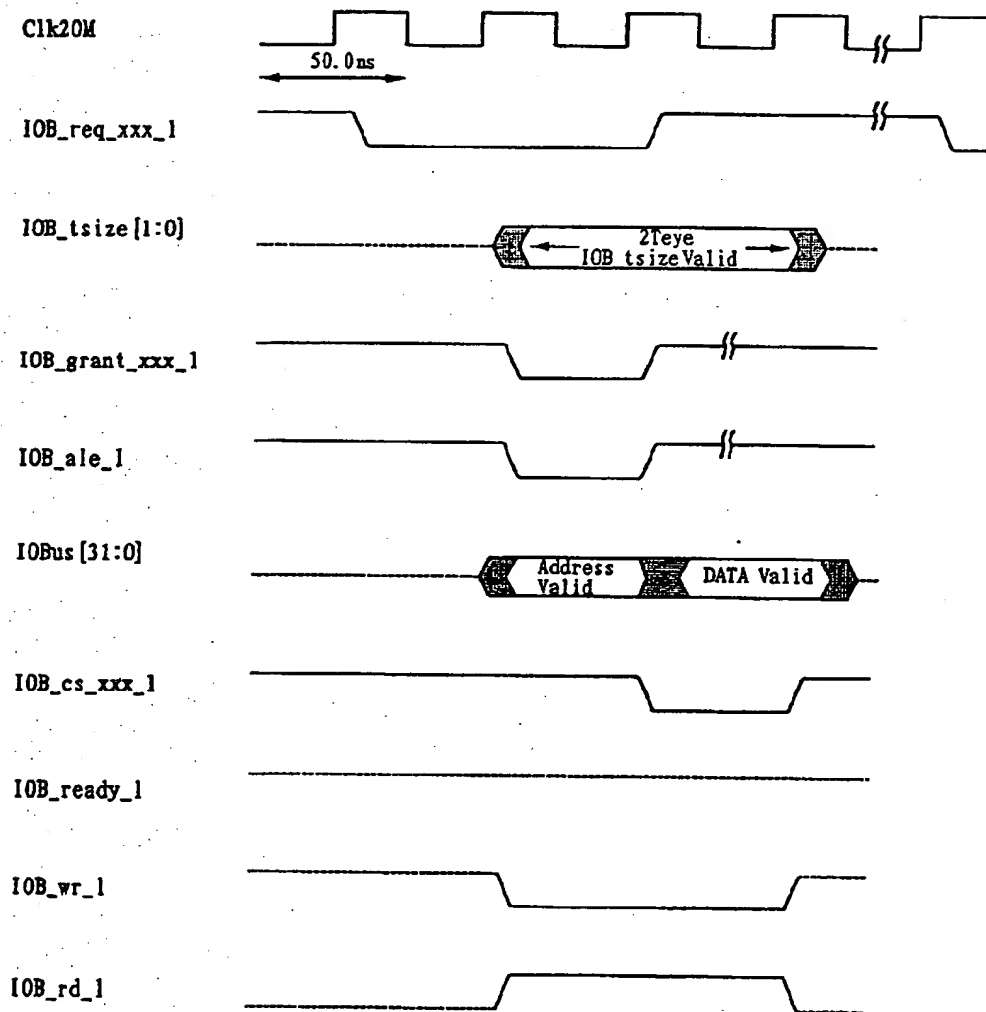
MSPメモリ制御器リクエストFIFOフォーマット

【図46】

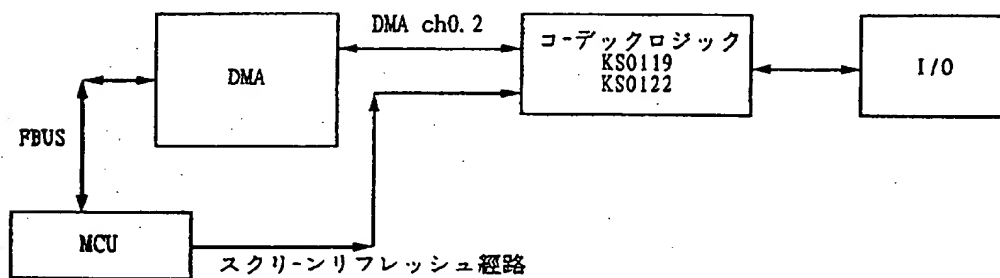


【図27】

IOBusライト (傳送サイズ=1ワード(4bytes))

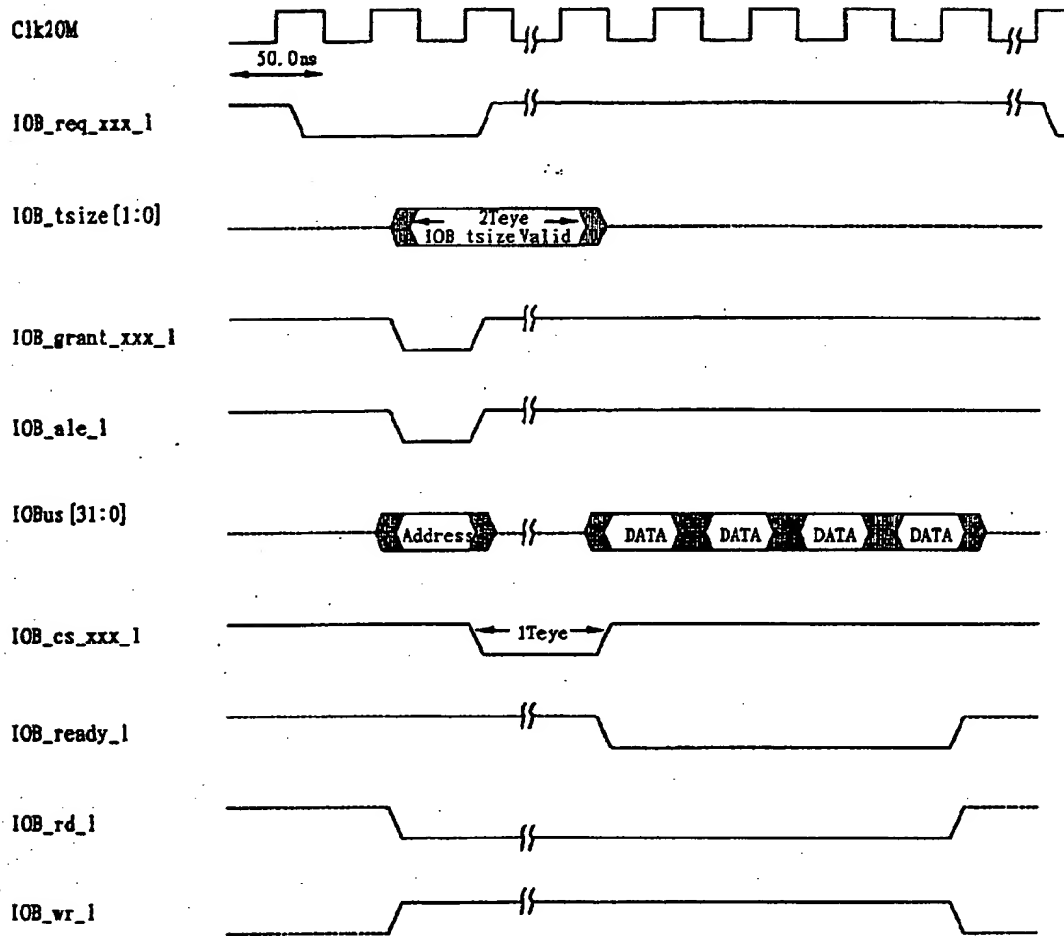


【図54】

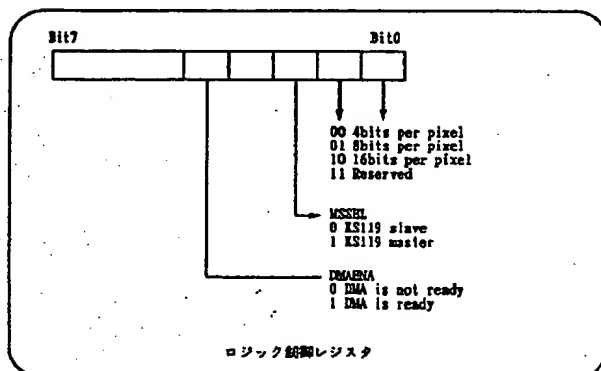


【図28】

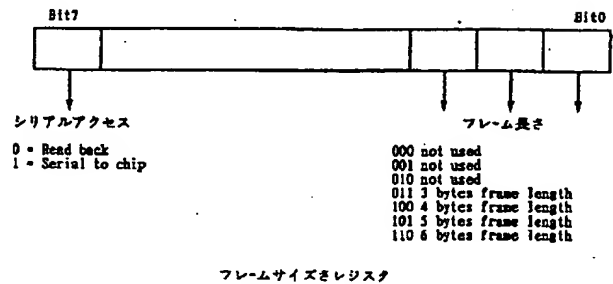
IOBus リード (傳送サイズ=4ワード (4bytes))



【図58】

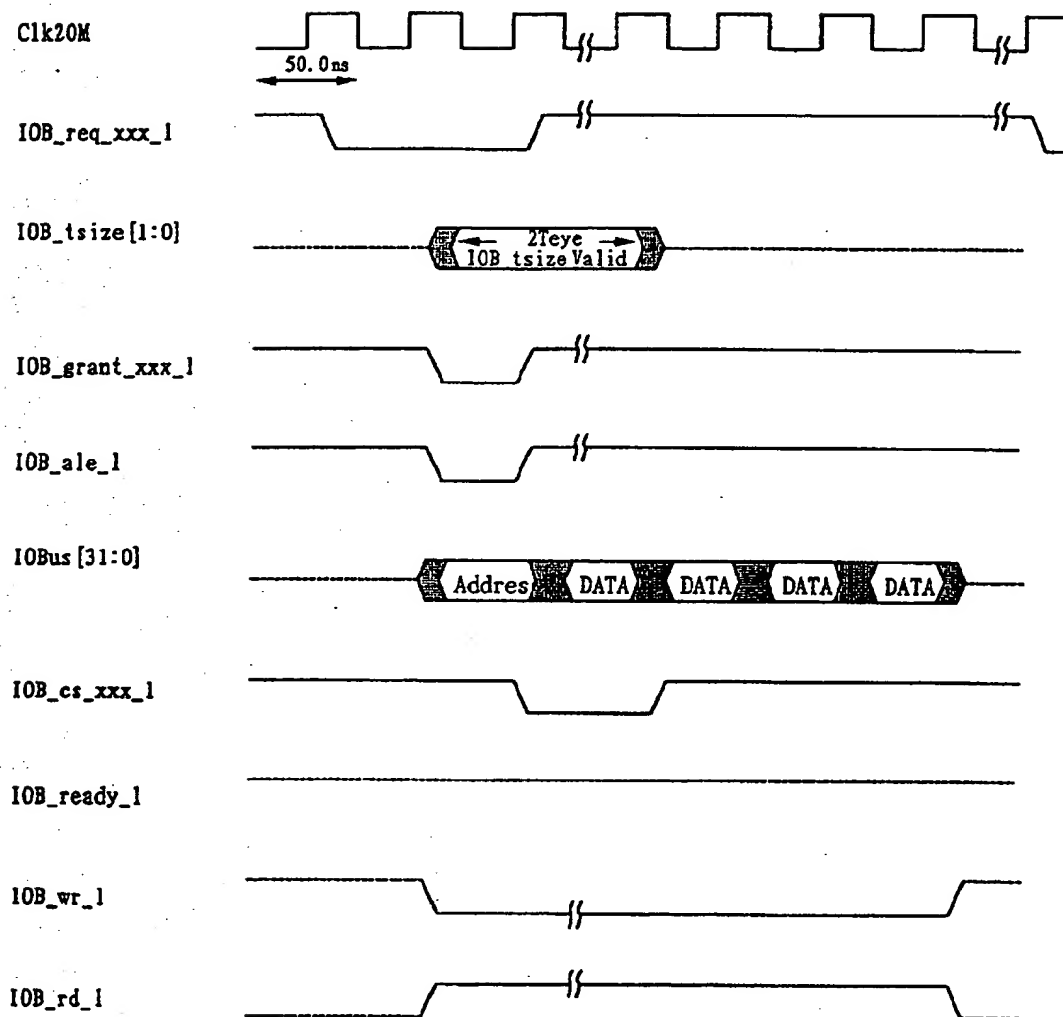


【図62】

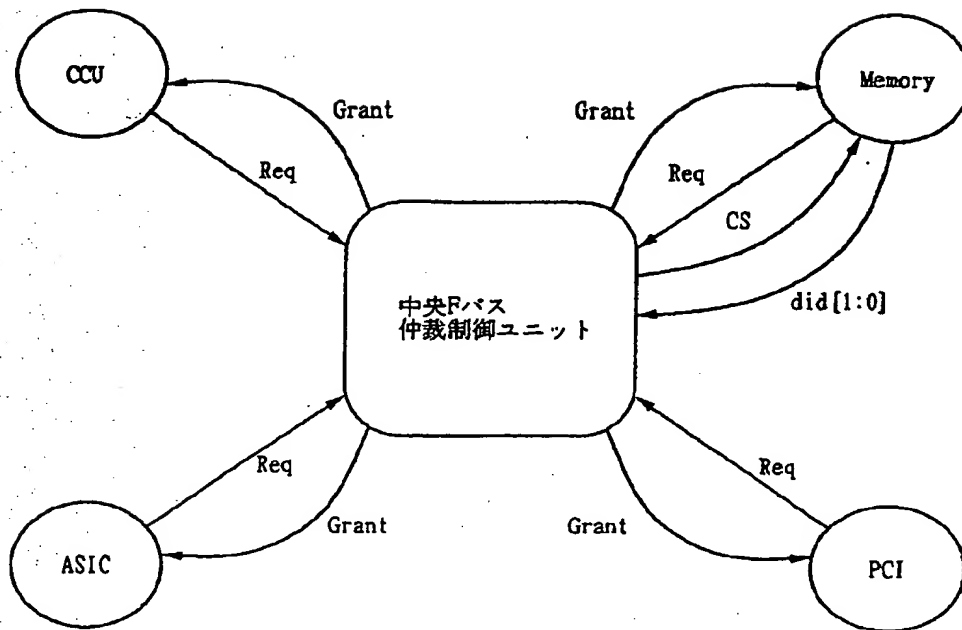


【図29】

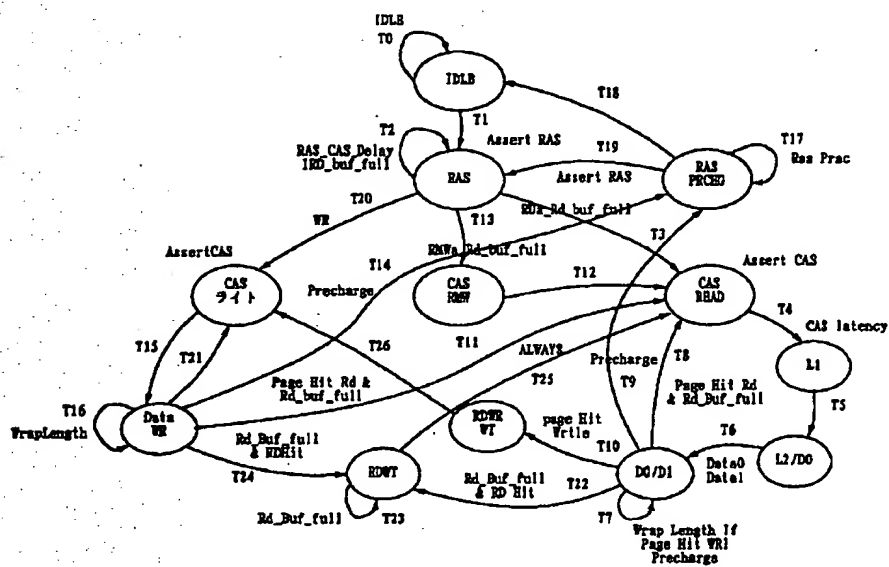
IOBusライト(傳送サイズ=4ワード(4bytes))



中央Fバス仲裁制御ユニット

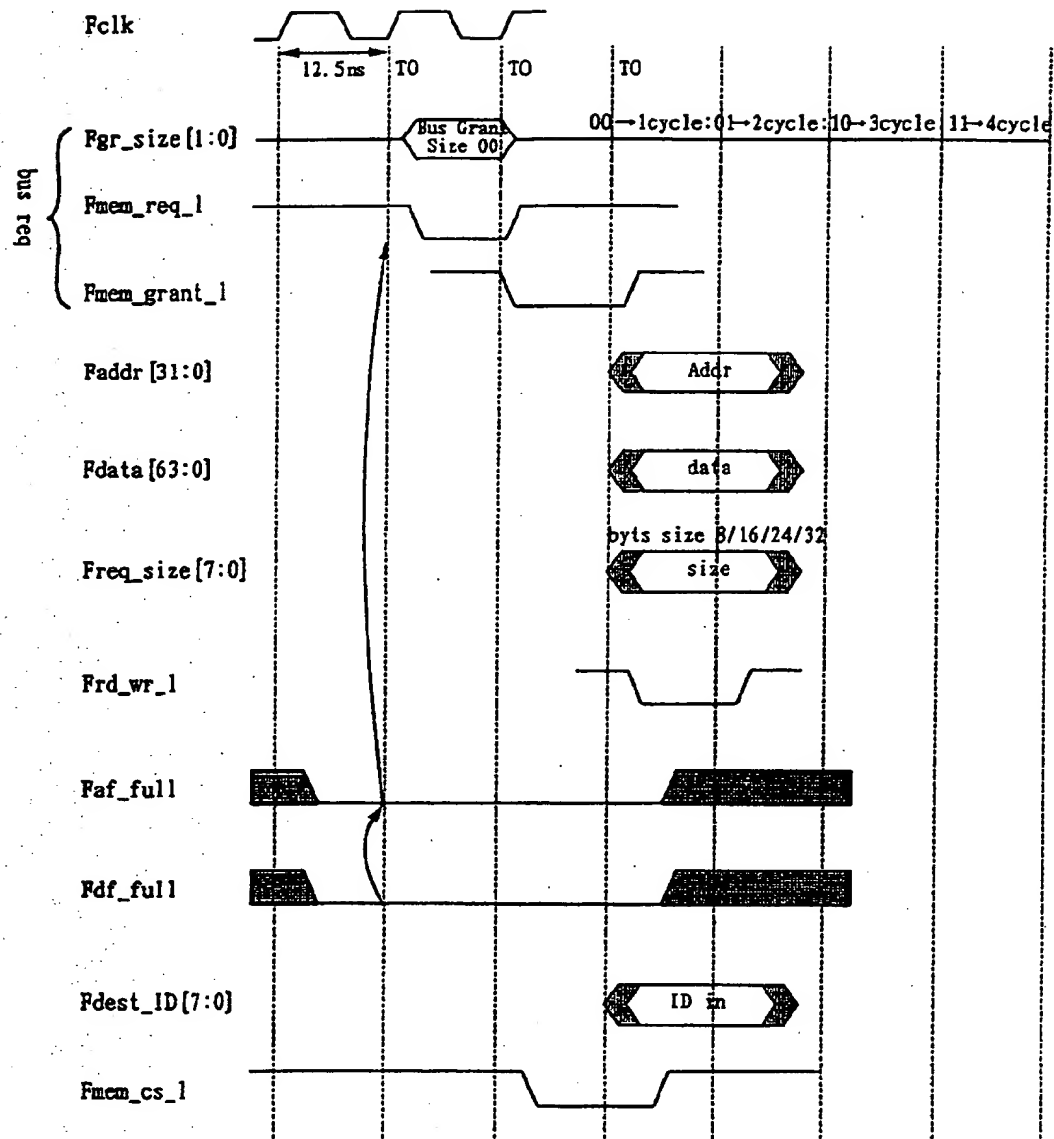


SDRAMメモリ制御器RAS/CAS状態マシン



【図34】

Fbusタイミング

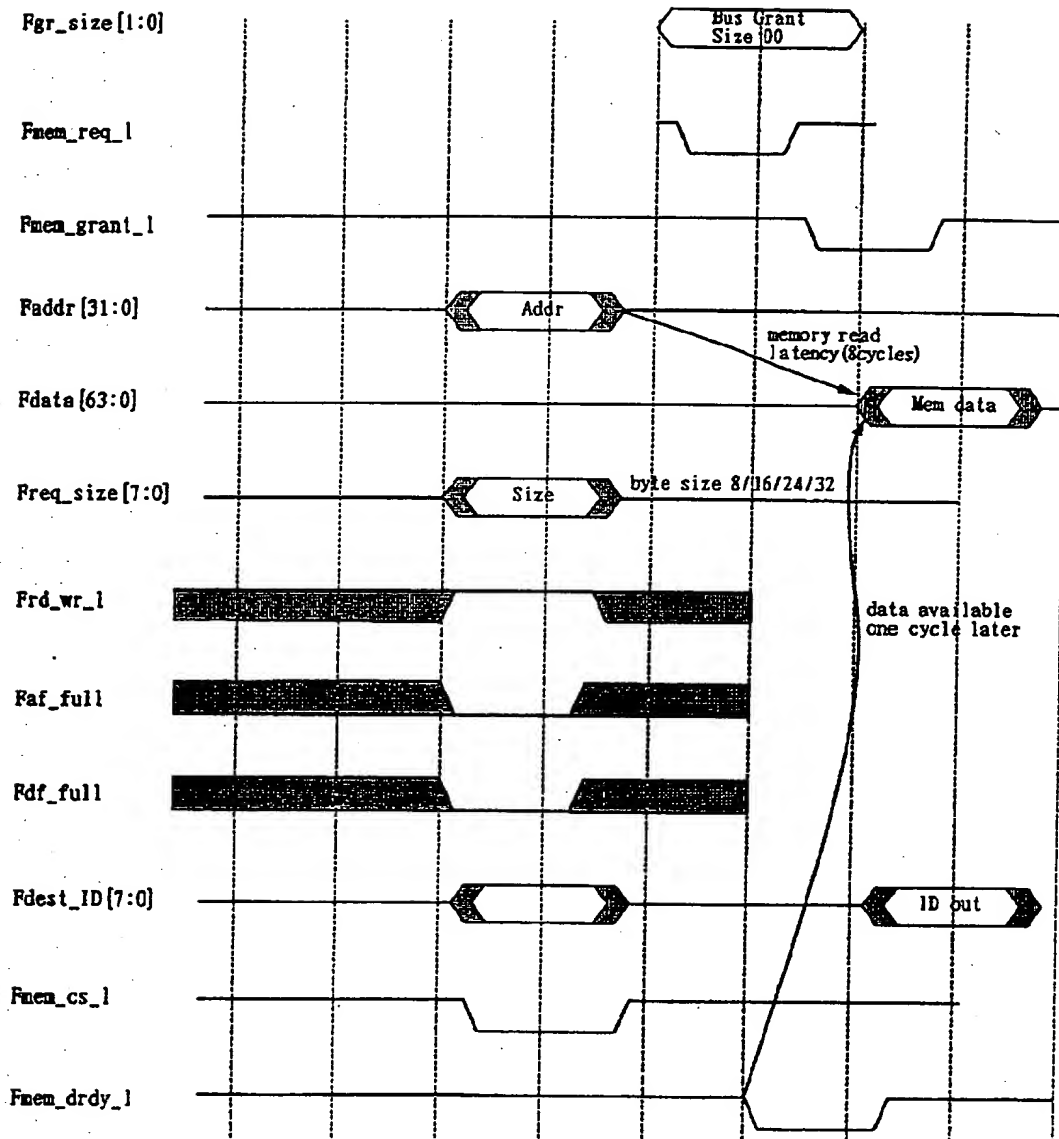


メモリ記入リクエストFbusタイミング

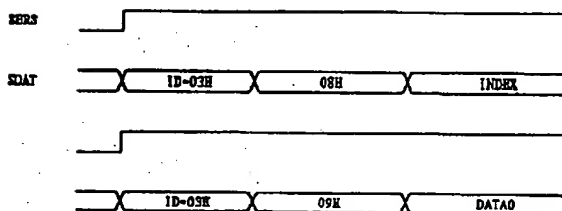
参照:ここには8バイトデータ伝送を示している.
16/32/64/128バイトに対しては
マルチプルデータサイクルが使用される.

【図35】

Fbusタイミング 図 cont.

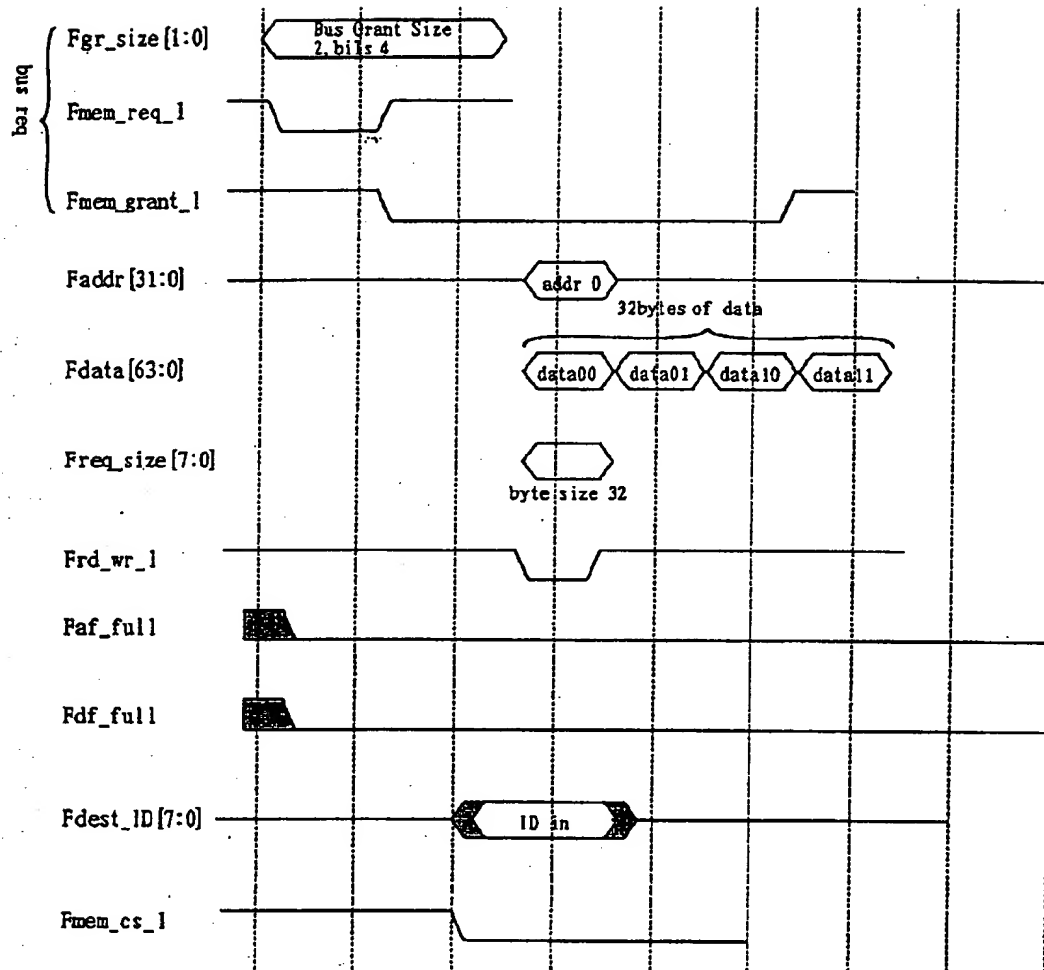
メモリ読取りクエストFbusタイミング
傳送サイズ=8バイト

【図63】



【図36】

Fbusタイミング 図 cont.

メモリ折り返し記入リクエスト
変換サイズ= 32バイト

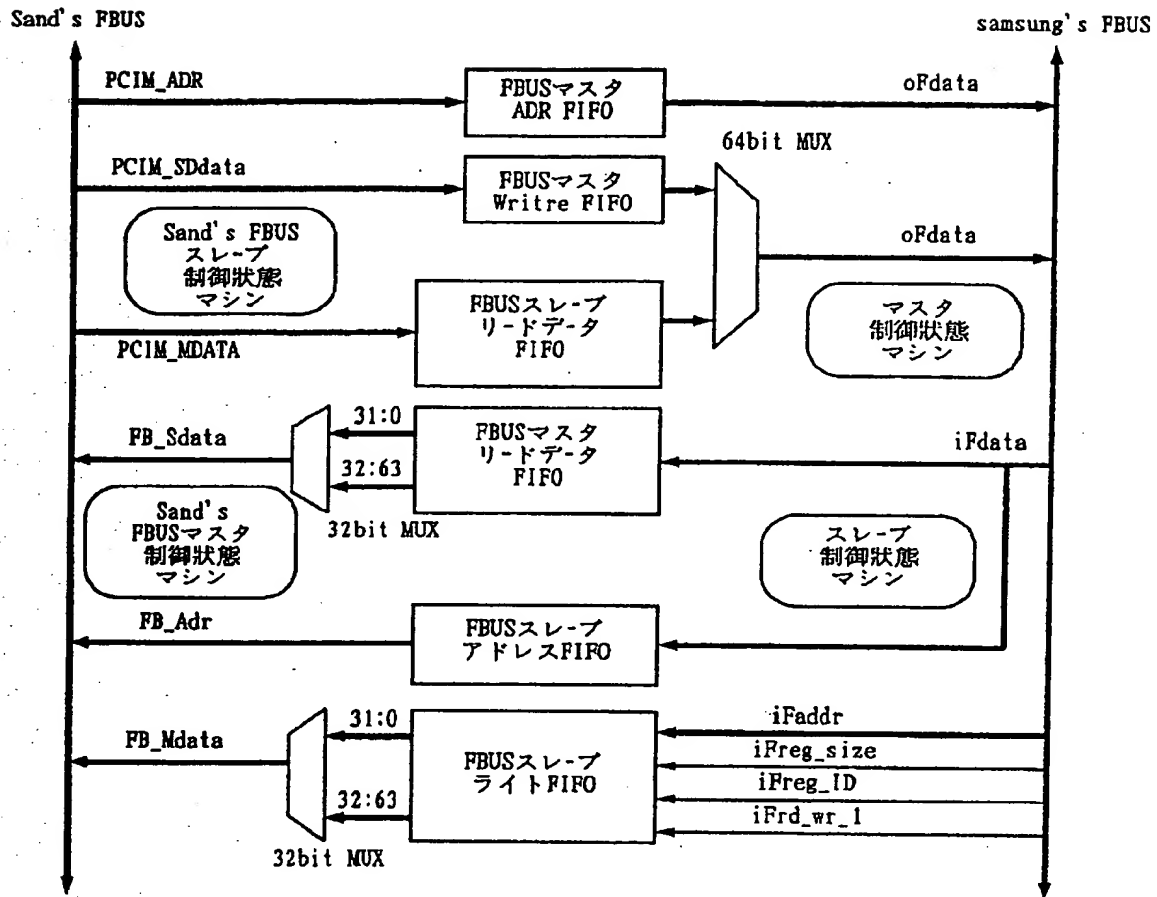
【図66】

デコーディング専用でビデオレートバッファに対する具現例

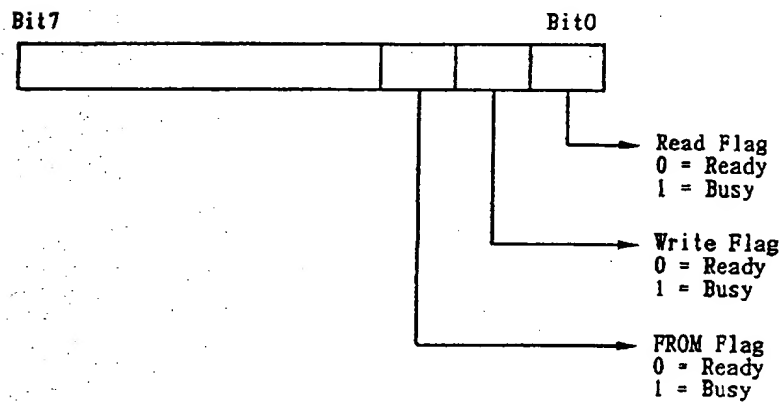
Rate_Buffer_Address_0	Rate_Buffer_Block_0
Rate_Buffer_Address_1	
Rate_Buffer_Address_2	Rate_Buffer_Block_1
Rate_Buffer_Address_3	
Rate_Buffer_Address_4	Rate_Buffer_Block_2
Rate_Buffer_Address_5	
Rate_Buffer_Address_6	Rate_Buffer_Block_3
Rate_Buffer_Address_7	

【図38】

FBUS PCIインターフェースロジック

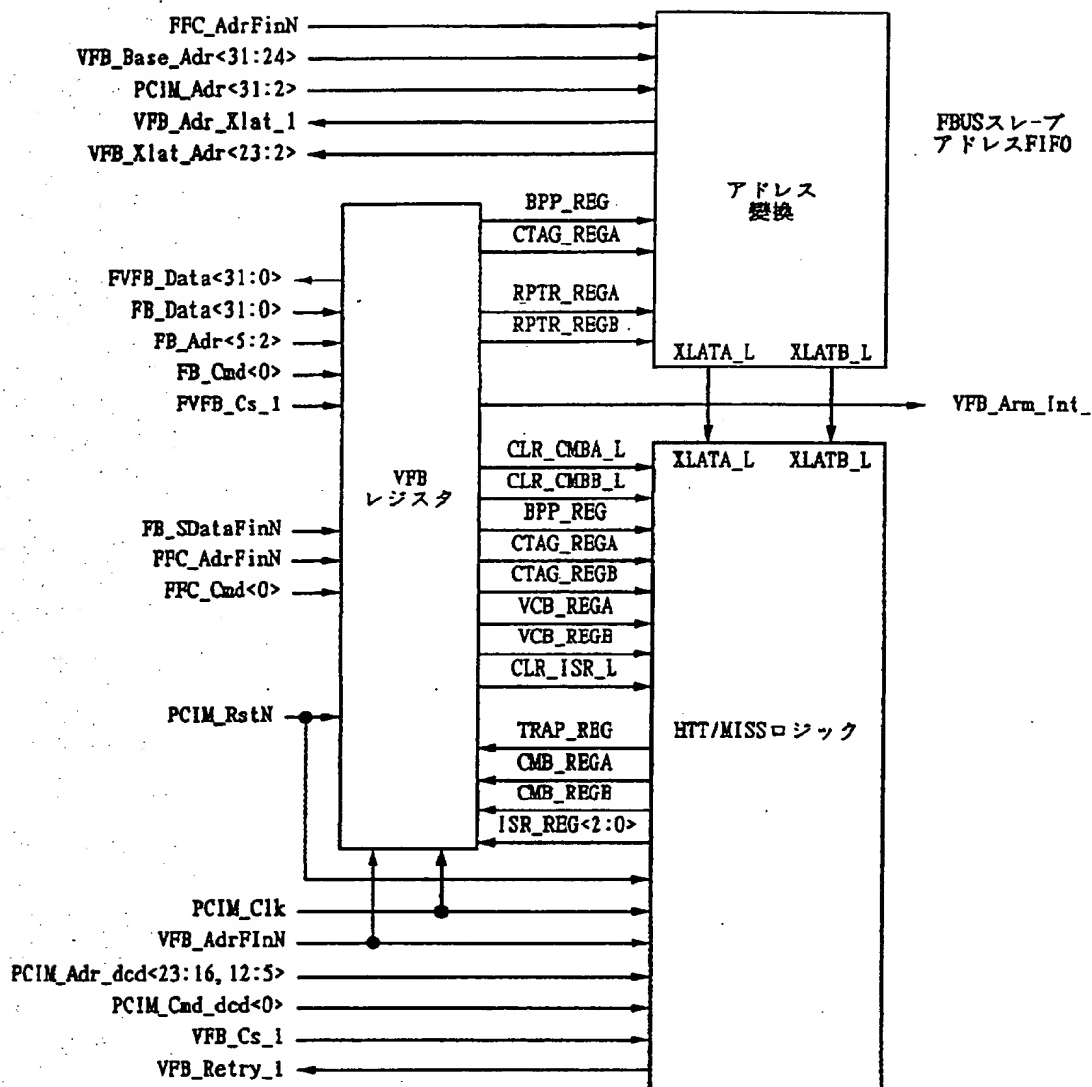


【図59】



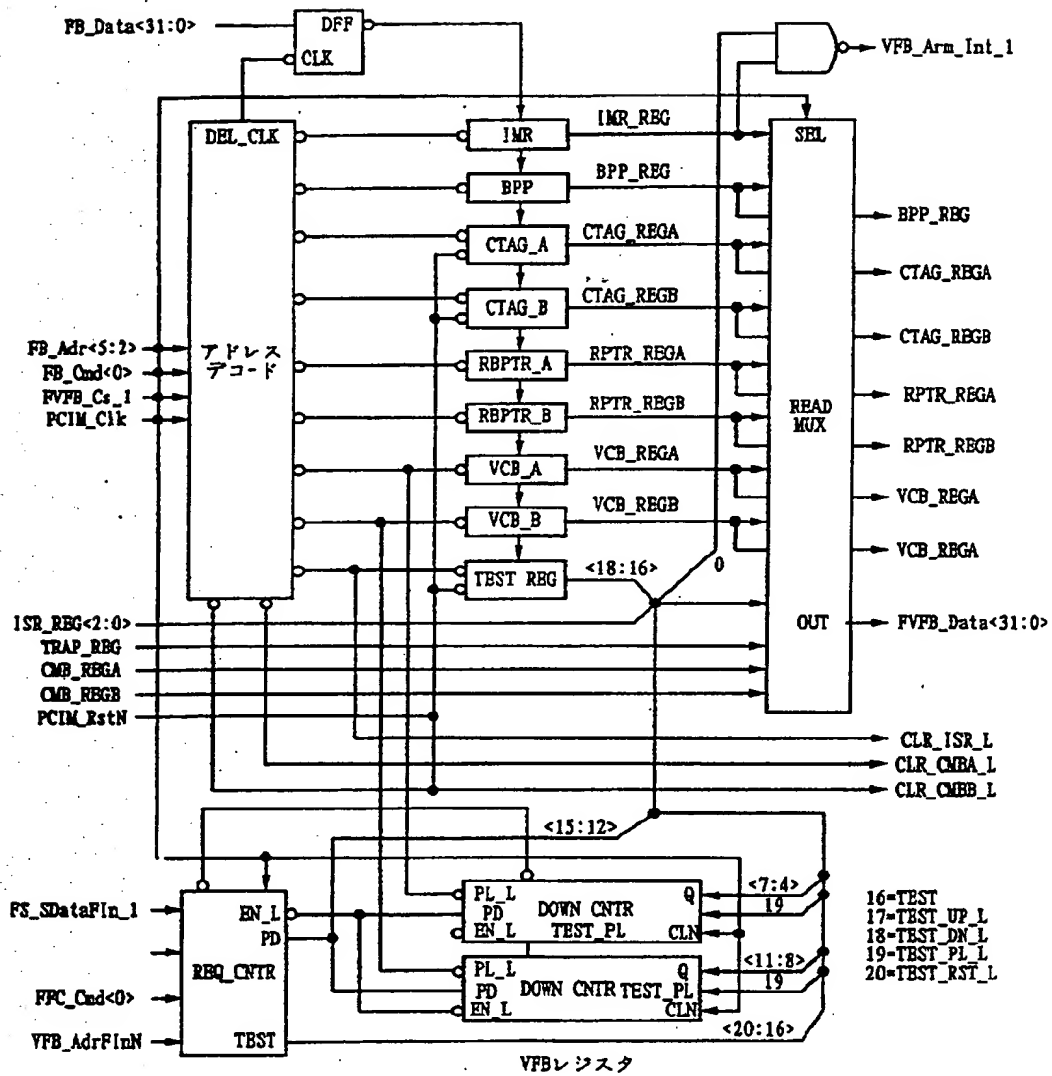
ステータスレジスタ

【図39】

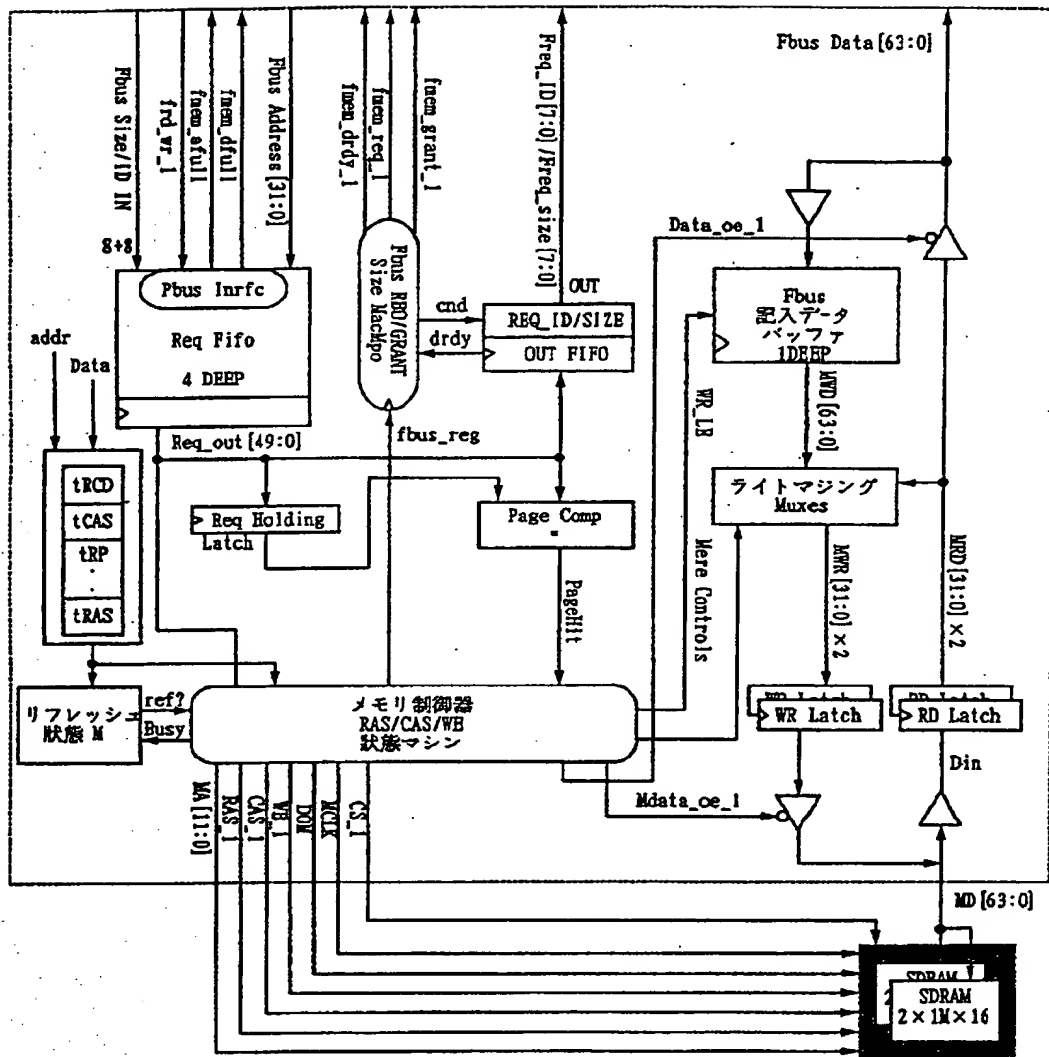


VFBブロック図

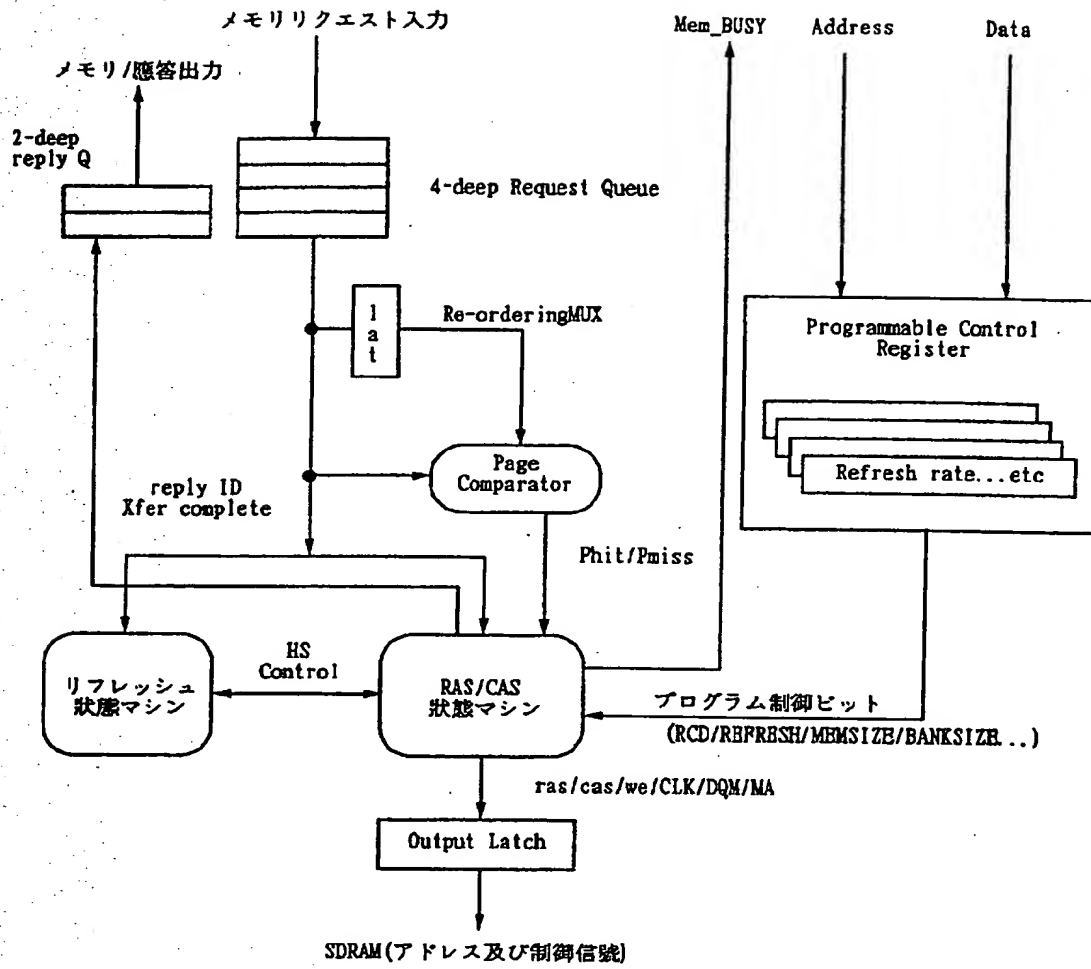
【図40】



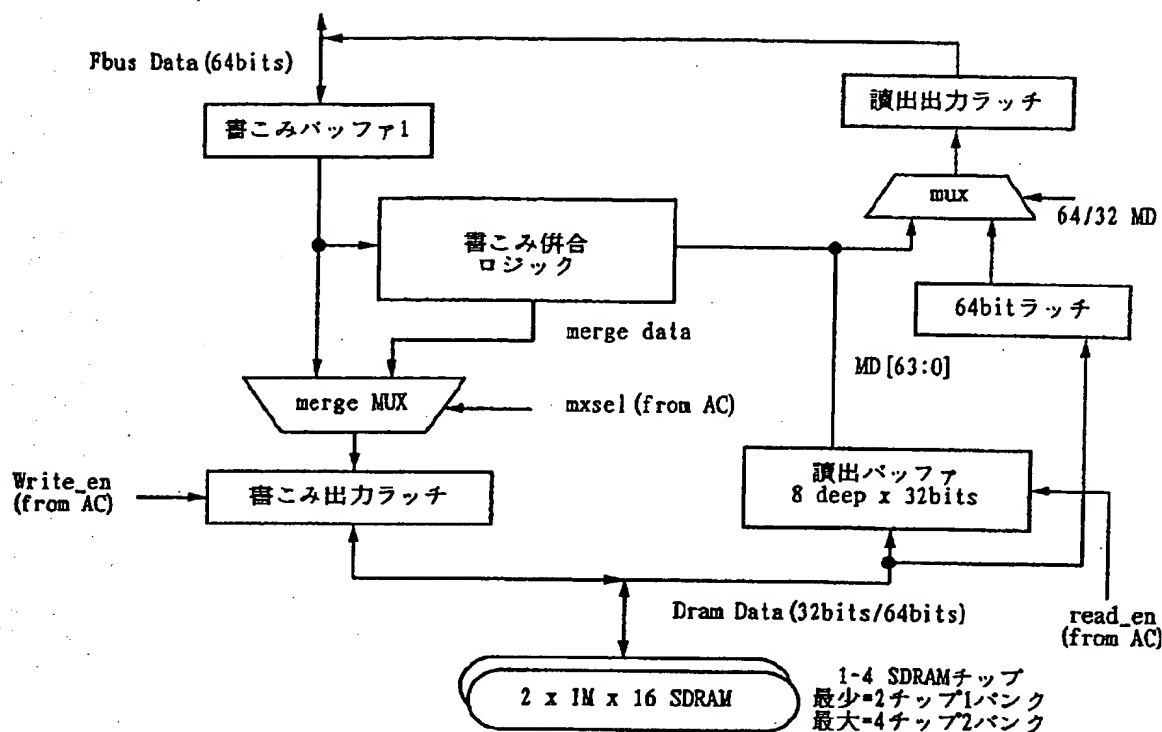
【図43】



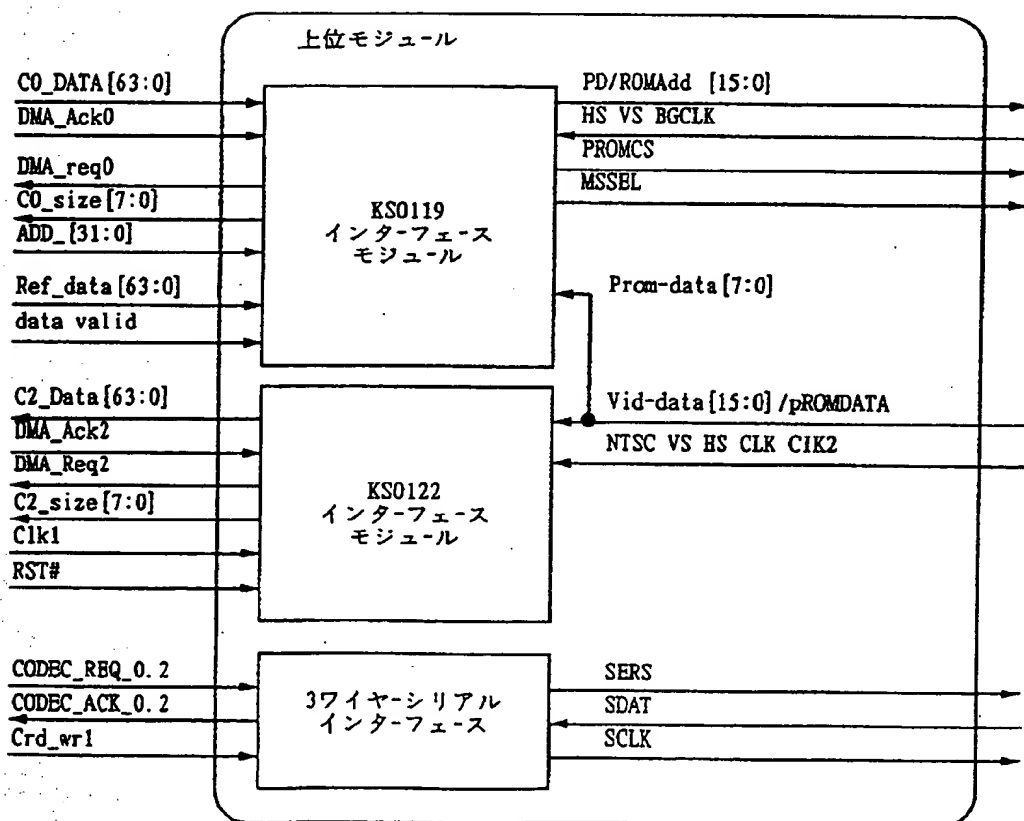
【図44】



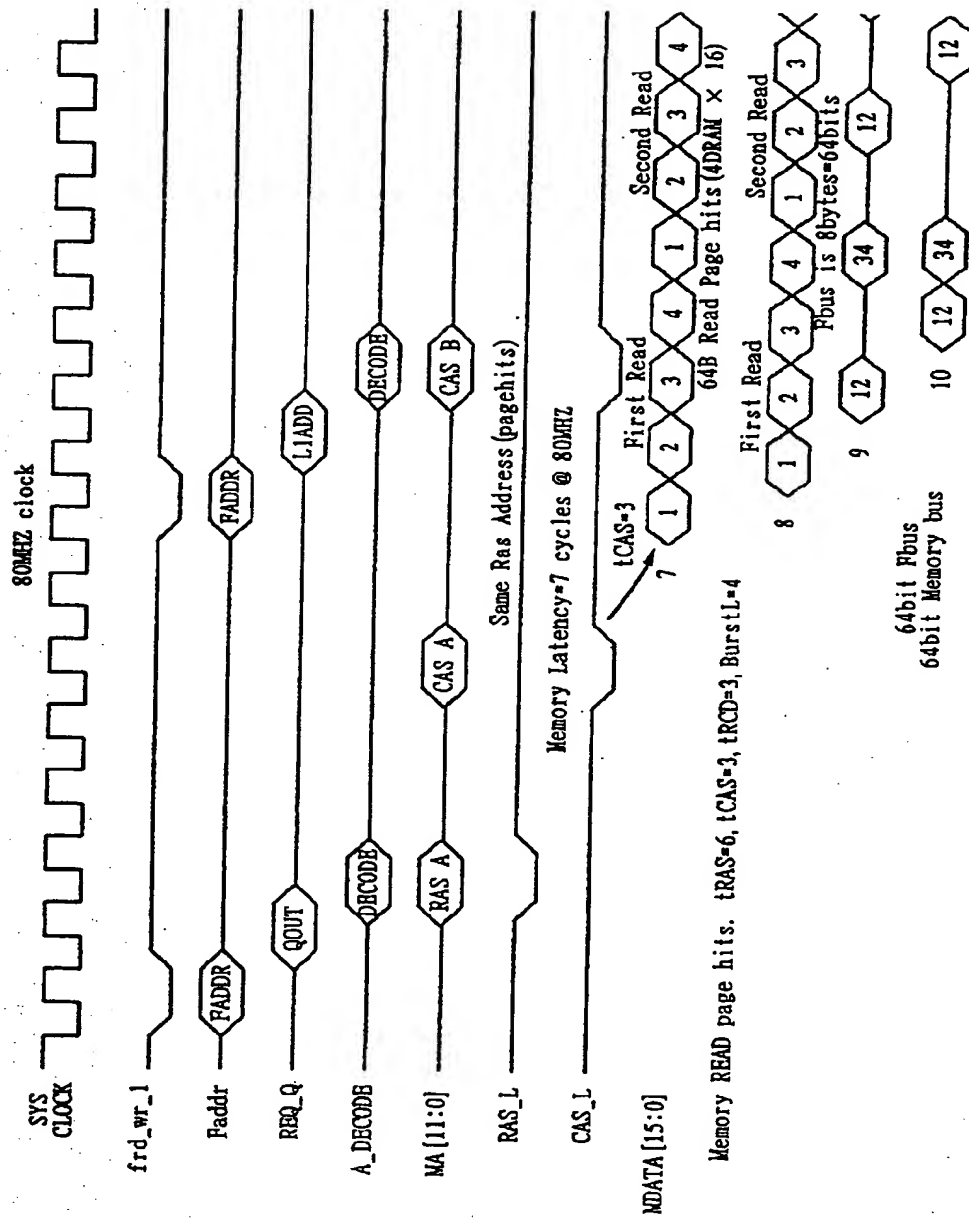
【図48】



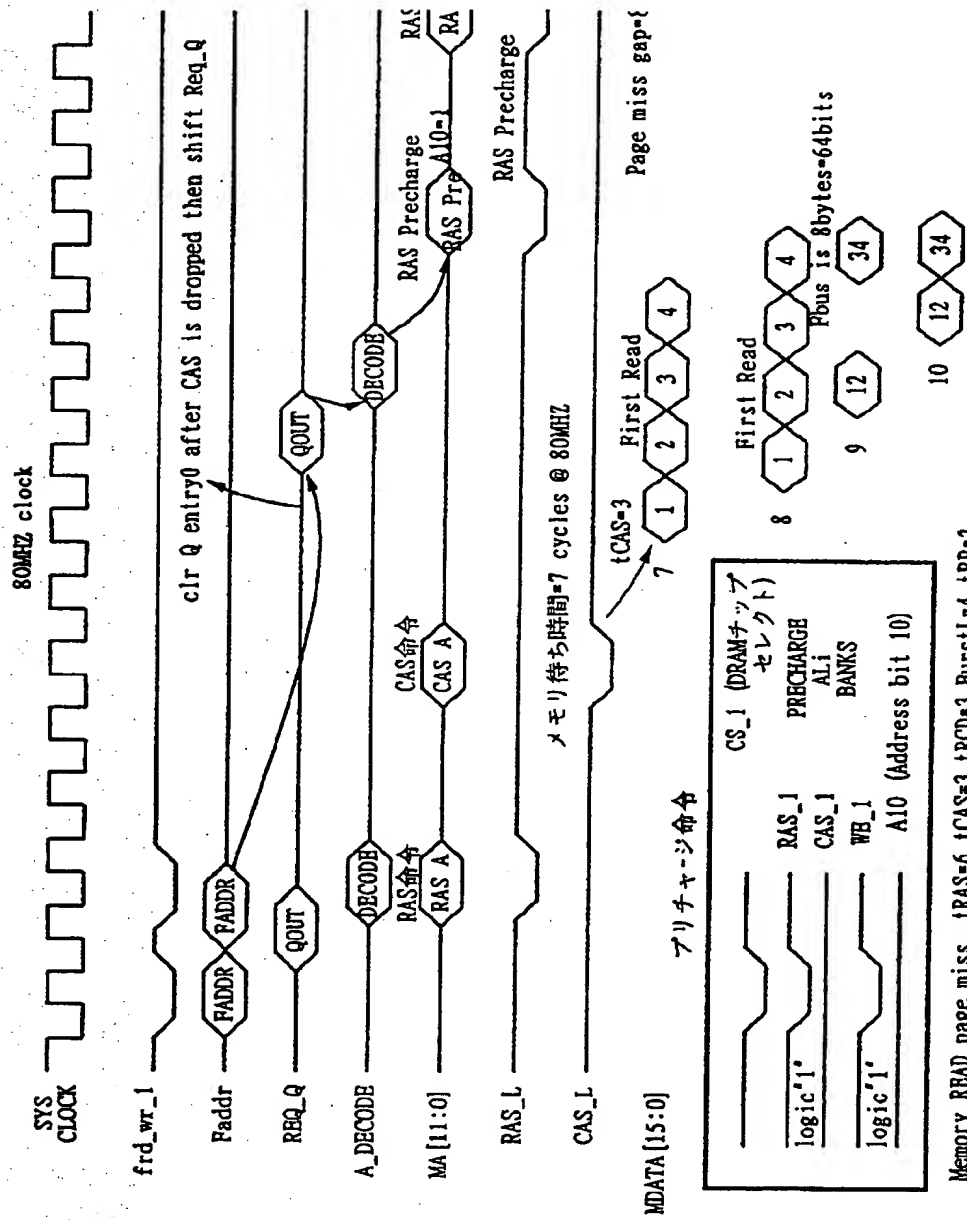
【図55】



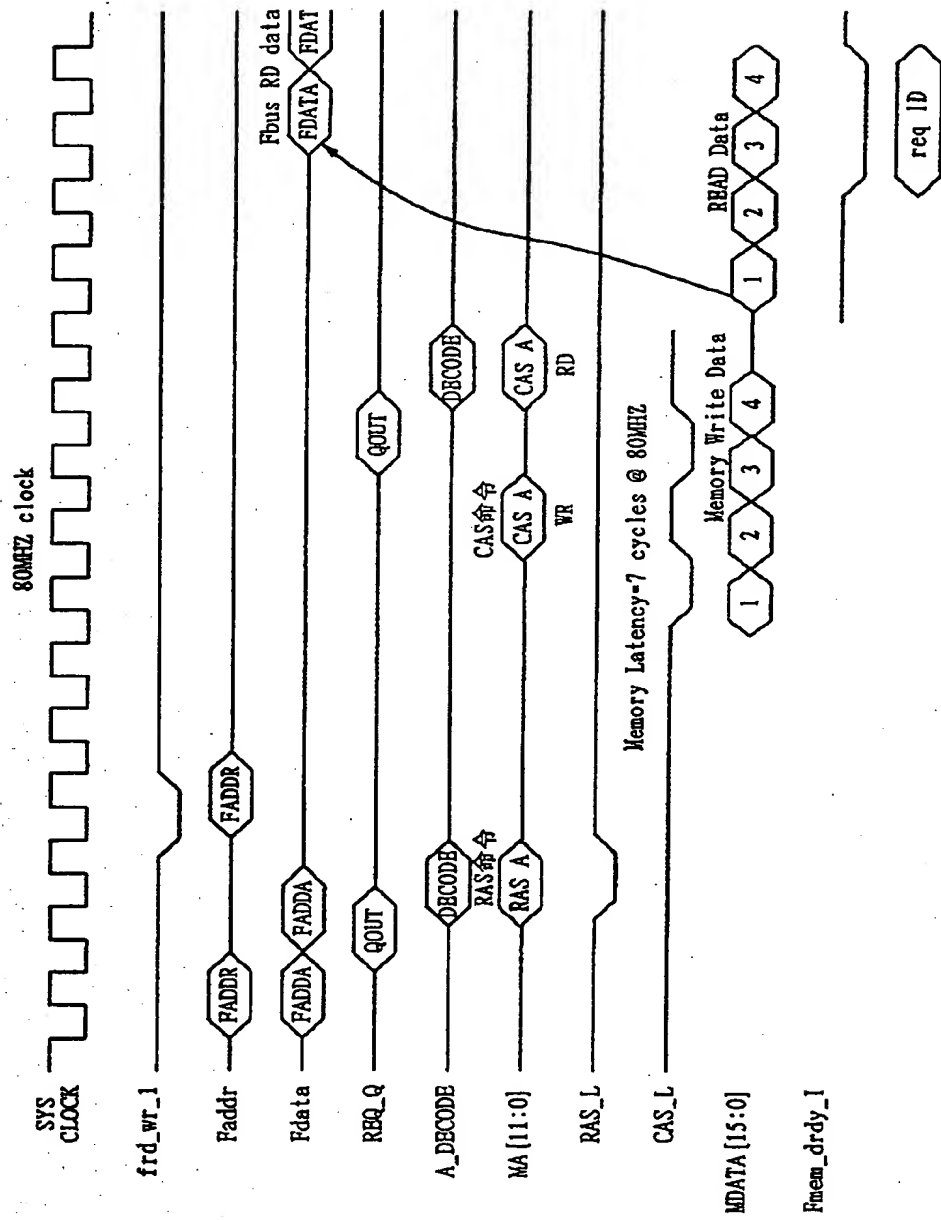
【図49】



【図 50】

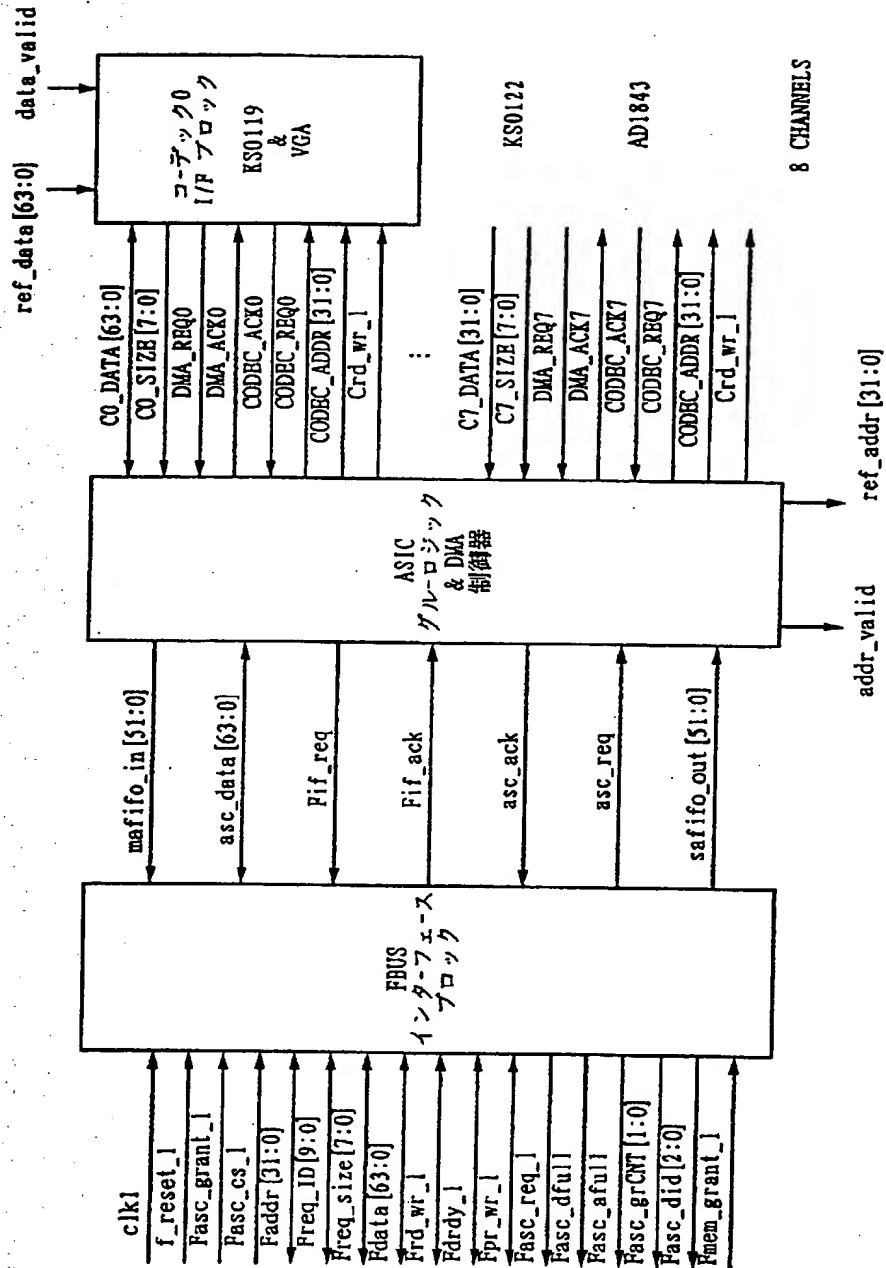


【図51】



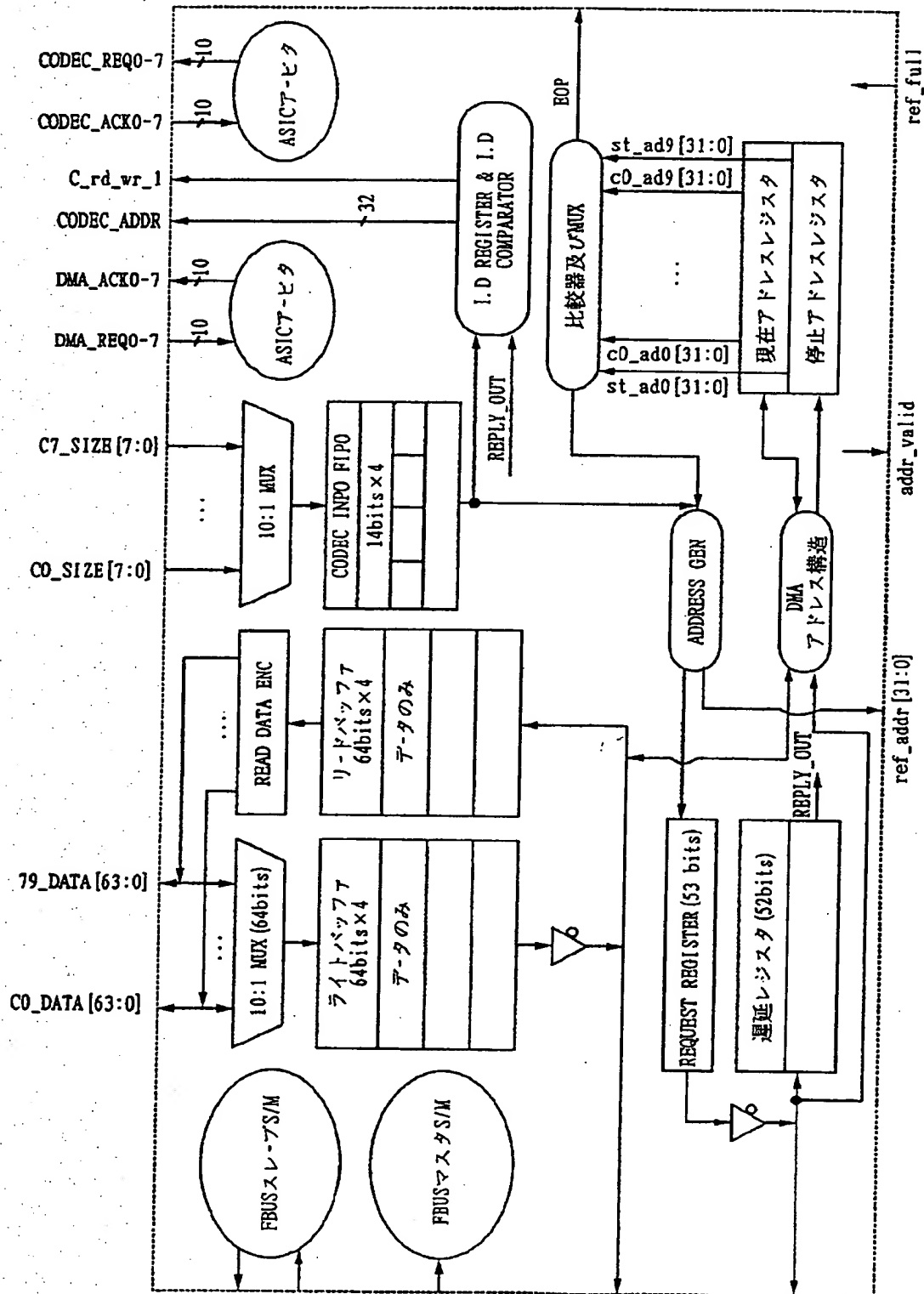
Memory READ page hits. tRAS=6, tCAS=3, tRCD=3, tRP=3

【図52】

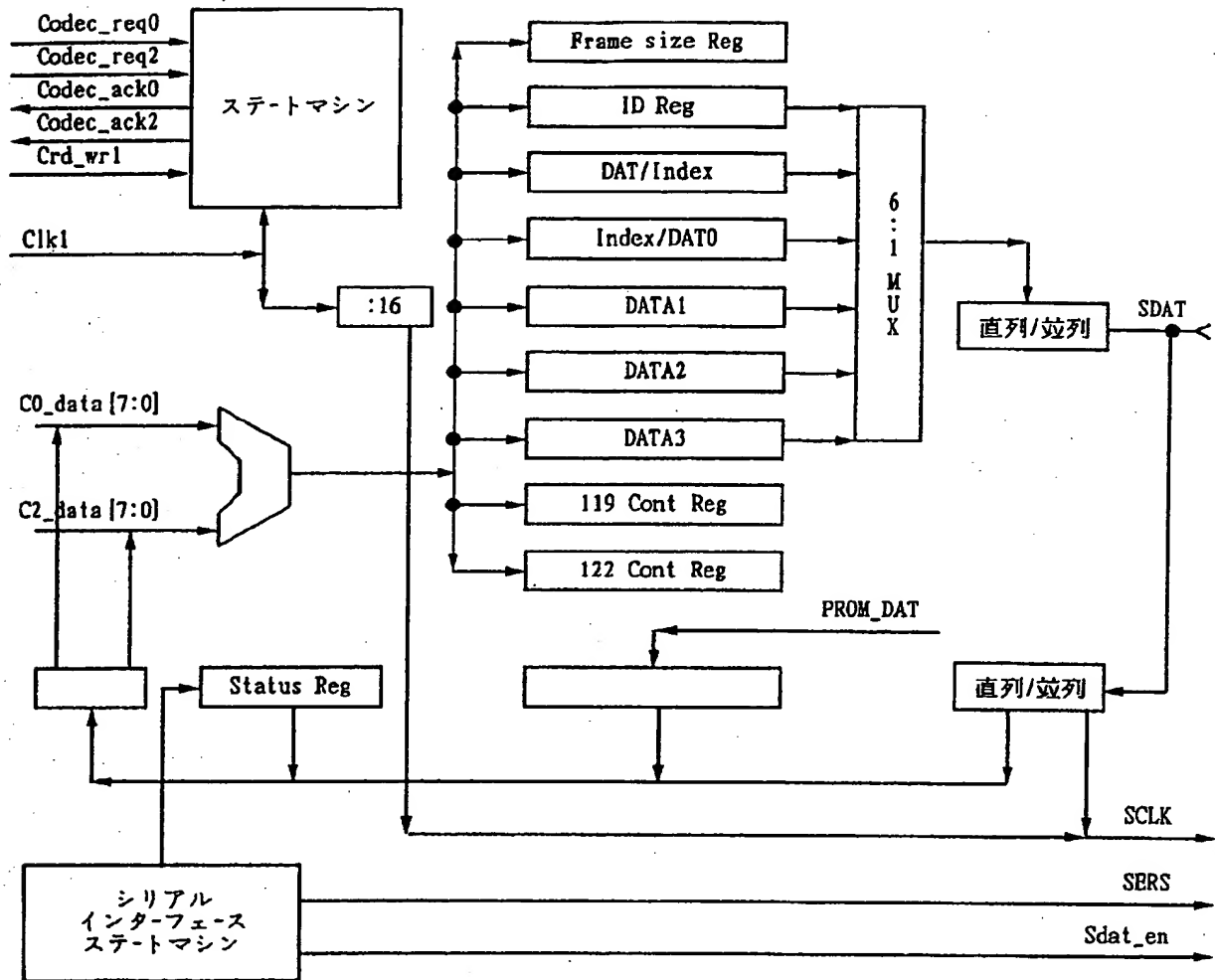


ASICインタフェースブロック

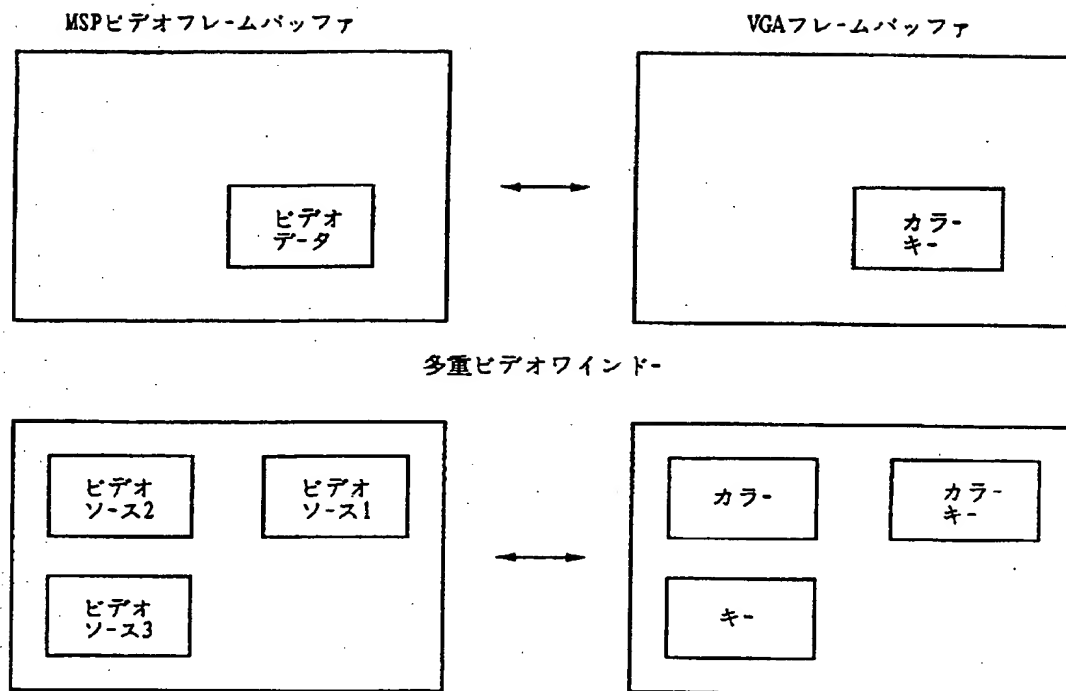
【図53】



【図56】



【図60】



フロントページの続き

(72)発明者 アムジャド・クレシ
 アメリカ合衆国・カリフォルニア・
 95134・サン・ホセ・イラン・ヴィレッ
 ジ・レーン・#307・350

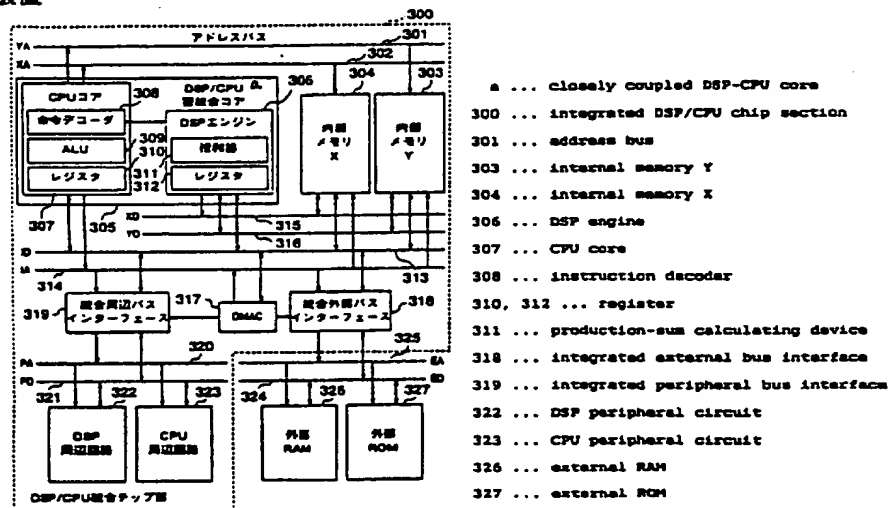
(72)発明者 ル・ングイエン
 アメリカ合衆国・カリフォルニア・
 95030・モンテ・セレノ・ダニエル・プレ
 イス・15095



(51) 国際特許分類 G06F 9/30	A1	(11) 国際公開番号 WO97/14093 (43) 国際公開日 1997年4月17日(17.04.97)
(21) 国際出願番号 PCT/JP96/02910 (22) 国際出願日 1996年10月7日(07.10.96) (30) 優先権データ 特願平7/261177 1995年10月9日(09.10.95) JP (71) 出願人 (米国を除くすべての指定国について) 株式会社 日立製作所(HITACHI, LTD.)(JP/JP) 〒101 東京都千代田区神田駿河台四丁目6番地 Tokyo, (JP) (72) 発明者：および (75) 発明者／出願人 (米国についてのみ) 中川哲也(NAKAGAWA, Tetsuya)(JP/JP) 〒184 東京都小金井市本町5丁目15-9 Tokyo, (JP) 波多野雄治(HATANO, Yuji)(JP/JP) 〒185 東京都国分寺市光町1丁目1-29 Tokyo, (JP) 堤坂康博(SAGESAKA, Yasuhiro)(JP/JP) 〒187 東京都小平市回田町219 Tokyo, (JP) 馬路 徹(BAJI, Toru)(JP/JP) 〒206 東京都稲城市向陽台4丁目2-C-608 Tokyo, (JP) 野口孝樹(NOGUCHI, Koki)(JP/JP) 〒156 東京都世田谷区上北沢3丁目23-18 Tokyo, (JP)		(74) 代理人 弁理士 浅村 皓, 外(ASAMURA, Kiyoshi et al.) 〒100 東京都千代田区大手町2丁目2番1号 新大手町ビル331 Tokyo, (JP) (81) 指定国 CN, JP, KR, SG, US, 欧州特許 (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). 添付公開書類 国際調査報告書

(54)Title: TERMINAL

(54)発明の名称 端末装置



(57) Abstract

For manufacturing a mobile communication terminal, reduction of the cost, power consumption, and size is a very important factor, and it is a major problem for the conventional technique in which two independent sets of DSPs and CPUs are used, because two systems of external memories are required. Further, since two systems of peripheral devices for data input/output are necessary for the DSPs and CPUs, there exists useless overhead between the DSPs and CPUs. A mobile communication terminal system is realized by using an integrated DSP/CPU chip having a DSP/CPU core (500) integrated as a single bus master, an integrated external bus interface (606) and an integrated peripheral circuit interface. Therefore, an inexpensive, low-power consumption, small-sized mobile communication terminal system is provided because the memory systems and peripheral circuits of the DSPs and CPUs are integrated.

(57) 要約

移動通信端末においては低コスト化、低消費電力化、小サイズ化の3つが非常に重要であるが、従来の2つの独立したDSPとCPUを用いた技術では2系統の外付けメモリが必要なため大きな問題となる。また、データ入出力用の周辺装置もDSP用とCPU用に2系統必要となりDSPとCPU間の余分な通信オーバーヘッドが存在していた。一つのバスマスターとして統合されたDSP/CPUコア(500)と統合された外部バスインタフェース(606)および統合された周辺回路インタフェースを持ったDSP/CPU統合チップで移動通信端末システムを実現する。DSPとCPUのメモリシステムと周辺回路が統合化することができ低コスト、低消費電力、小サイズの移動通信端末システムを実現できる。

情報としての用途のみ

PCTに基づいて公開される国際出願をパンフレット第一頁にPCT加盟国を特定するために使用されるコード

AL	アルバニア	ES	スペイン	LR	リベリア	RU	ロシア
AM	アルメニア	FI	フィンランド	LS	レソト	RG	ルーマニア
AT	オーストリア	FR	フランス	LT	リトアニア	SS	スウェーデン
AU	オーストラリア	GB	イギリス	LV	ラトヴィア	SI	スロベニア
AZ	アゼルバイジャン	GG	ガブリ	MC	モナコ	SK	スロバキア
BB	バハマ	GH	ガーナ	MD	モルドバ	SN	セネガル
BE	ベルギー	HN	ホンジュラス	MG	マダガスカル	SO	ソマリア
BF	ブルキナファソ	IE	アイルランド	MK	マケドニア	TD	チャド
BG	ブルガリア	IT	イタリア	ML	マリ	TG	トゴ
BR	ブラジル	JP	日本	MN	モンゴル	TM	トルクメニスタン
BS	バハマ	KE	ケニア	MR	モーリタニア	TT	トリニダード・トバゴ
BT	ブータン	KR	韓国	MW	マラウイ	TV	ツバル
BZ	ベリーズ	KG	キルギス	MX	メキシコ	UA	ウクライナ
CA	カナダ	KK	カザフスタン	NE	ニジェール	UG	ウガンダ
CC	ココス(キリング)諸島	KL	マレーシア	NL	オランダ	US	アメリカ合衆国
CD	コンゴ民主共和国	LI	リヒテンシュタイン	NZ	ニュージーランド	UY	ウルグアイ
CF	中央アフリカ共和国			PT	ポルトガル		
CG	コンゴ共和国			RO	ルーマニア		
CH	スイス						
CN	中国						
CO	コロンビア						
CR	コスタリカ						
CU	キューバ						
DE	ドイツ						
DK	デンマーク						